Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1983-03

A computer simulation of a combat model which uses command and control.

Bent, James E. III

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/19695

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A COMPUTER SIMULATION OF
A COMBAT MODEL WHICH USES
COMMAND AND CONTROL

by

James E. Bent III

March 1983

Thesis Advisor: D. P. Gaver

Approved for public release; distribution unlimited.

hesis
3917

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A COMPUTER SIMULATION OF A COMBAT MODEL WHICH USES COMMAND AND CONTROL

by

James E. Bent III

March 1983

Thesis Advisor: D. P. Gaver

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Computer Simulation of a Combat Model Which Uses Command and Control		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; March 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James E. Bent III		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1983
		13. NUMBER OF PAGES 91
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Command and Control
Combat Simulation
Interactive Graphics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis provides a student of Command and Control with a computer simulation of a simple Command and Control (C2) model. The simulation is a user-friendly, interactive program with multi-colored, high-resolution graphical displays to illustrate the effect of C2 on a stylized, simple,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

combat situation. A User's Manual is provided to facilitate the use of the simulation.

Approved for public release; distribution unlimited.

A Computer Simulation of a Combat Model which uses Command and Control

by

James E. Bent III
Captain, United States Air Force
B.S., North Texas State University, 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL AND COMMUNICATIONS)

from the

NAVAL POSTGRADUATE SCHOOL
March 1983

B3
c.2

ABSTRACT

This thesis provides a student of Command and Control with a computer simulation of a simple Command and Control (C2) model. The simulation is a user-friendly, interactive program with multi-colored, high-resolution graphical displays to illustrate the effect of C2 on a stylized, simple, combat situation. A User's Manual is provided to facilitate the use of the simulation.

TABLE OF CONTENTS

I. INTRODUCTION 9

II. THE SIMULATION 11

A. GENERAL 11

B. THE PLAYING BOARD 12

C. THE PLAYING PIECES 12

D. SEQUENCE OF PLAY 15

1. The User Input 15

2. Selection of the Red Forces' Route of Attack 18

3. Phase Three 20

4. After-Battle Analysis 23

III. A SIMPLE C2 MODEL 32

IV. IMPLEMENTATION OF THE C2 MODEL 36

A. THE OPPOSING FORCES 36

B. SIMULATING THE C2 PROCESS 37

C. THE HOSTILE FORCES 45

D. PRE-BATTLE DECISIONS 45

V. THE COMBAT MODEL 50

A. THE ATTRITION CALCULATIONS 50

1. Unaimed Attrition Calculation 50
2. Aimed Attrition Calculation 52

B. DETERMINISTIC VS STOCHASTIC VERSIONS 54

C. HAND-TO-HAND COMBAT 55

D. BATTLE TERMINATION 57

E. BREAK POINTS 57

F. FORCE MOVEMENT 57

VI. FUTURE ENHANCEMENTS 61

A. TWO-PLAYER GAME 61

B. RE-SUPPLY OF FIGHTING FORCES 62

C. NEW TRANSFORMATION EQUATIONS 62

D. DEVELOPMENT OF COMBAT SCENARIOS 63

APPENDIX A: USER'S MANUAL 65

A. INTRODUCTION 65

B. PROGRAM STRUCTURE 65

1. The Main Program 66
2. The Input Section 68
3. The Combat Section 68
4. The Command and Control Section 69
5. The Graphical Output Section 70

C. THE INPUT DATA 71

D. OUTPUT 79

1. Playing Board Displays 80

2. Force Strength vs Time Plots 80

E. TO RUN THE SIMULATION 82

APPENDIX B: 'DERIVATION OF ATTRITION EQUATIONS' 84

APPENDIX C: DERIVATION OF THE STOCHASTIC ATTRITION RATE 87

BIBLIOGRAPHY 90

INITIAL DISTRIBUTION LIST 91

LIST OF FIGURES

2.1 The Playing Board Layout 14

2.2 Table of Variables and Current Values 17

2.3 Hand-to-Hand Combat Range Explanation 19

2.4 User Inputting the Red Force Route of Attack . . 21

2.5 Combat Simulation Showing Red Force Status . . . 26

2.6 Combat Simulation Showing Blue Force Status . . 27

2.7 Combat Simulation Showing BC Outpost Status . . 28

2.8 Combat Simulation Showing Range Between Forces . 29

2.9 Example 1 of a Force Strength vs Time Plot . . . 30

2.10 Example 2 of a Force Strength vs Time Plot . . . 31

3.1 The C2 Process 33

3.2 The Time Components of the C2 Process 35

5.1 Red Movement Equations 60

A.1 Flow Chart of the Main Program 67

I. INTRODUCTION

There are many combat simulations available at the Naval Postgraduate School, but few, if any, have a Command and Control (C2) system as an integral part of the simulation. Those computer simulations which do employ a C2 system are often complicated and difficult to analyze. The purpose of this thesis is to develop a simple C2 model, and a simulation of the model, to demonstrate the effect of a Command and Control system on a stylized, simple, combat situation.

A highly user-friendly, interactive computer simulation is used to graphically illustrate the effect of C2 on combat. To enhance the interactive aspects of the simulation, a RAMTEK 9400 color graphics system having a high resolution graphics capability is used to display intermediate and end-of-battle results. All of the program's input variables are explained during the simulation so that once a user starts the program, he does not need a user's manual, although one is provided.

It is hoped that use of this simulation will provide a user with insights into the effect of C2 upon the outcome of conflicts. The user, by varying the amount of C2 and

counter C2 used during a particular battle, should come to understand and appreciate the importance of C2, and how C2 decisions can easily change the outcome of a battle.

This paper explains a simple C2 model and how the model is implemented in a computer simulation. The attrition and movement of forces are also explained to aid the user in understanding the combat portion of the simulation. Photograghs are provided to illustrate both the input and output of the program. Finally, a user's manual (Appendix A) is provided. The user's manual provides the documentation needed to run the program and to add new subroutines to enhance the computer simulation.

II. THE SIMULATION

A. GENERAL

The simulation uses a computer generated, color graphics playing board which represents a section of land. The board is divided into spaces (called hexes) in order to position the playing pieces and to regulate their movement. The playing pieces called units are made up of Red and Blue forces. The stylized scenario pits a Blue defending force against an advancing Red force. At the beginning of the simulation, the user initializes all of the input variables' values (approximately 40), determines the route the Red units will move along in their attack on the Blue units and then starts the conflict. At the end of the conflict, the user can display the battle results in a Force vs Time graph. After examining the end-of-battle results, the user may run the simulation again with the same initial values for all the variables, or changes may be made to any variable desired. The user can then run the simulation to determine if the outcome of the conflict would have changed as a result of changes in the initial value of the variables.

B. THE PLAYING BOARD

The computer generated, high-resolution color graphics playing board represents a piece of ground. It is divided into 68 hexes to regulate the movement and location of the units. There are three types of hexes shown on the board (see Figure 2.1). Hexes 26,33, 37, 56, and 63 are green in color and represent forest, or woods. Hexes 17, 32, and 45 are magenta and represent towns or cities. The remaining hexes are yellow and represent ordinary land (fields).

In the present version of the simulation the cities and forests have no effect on the forces or the simulation results. However, it is envisioned that combat situations could be developed incorporating these features to allow the user to explore the effect of these features on command and control.

C. THE PLAYING PIECES

There are two types of forces, Red and Blue, used in the simulation. The Blue forces which can be composed of three types of units (see Figure 2.1)

*Blue unaimed units represented on the board by the blue letter U,

*Blue sensor-aimed units represented on the board by the blue letter A, and

*Blue command and control outposts represented on the board by the blue letters BC.

The Red forces, on the other hand, can be divided into two types of units (see Figure 2.1)

*Red Main units represented on the board by the red letter M, and

*Red counter-command and control units represented on the board by the red letter C.

The letters representing the different units can have any of five different sizes during the course of the battle. At the beginning of the simulation all the letters, except for the blue letter A, are full-sized, indicating the units are at full strength. As the unit's strength is depleted (thru attrition) by decrements of 20% of the initial unit strength, the size of the letter decreases. Thus in Figure 2.1, the red letter C indicates that the Red counter-command and control unit has lost over 40% of its initial strength while the blue letter U indicates that the Blue unaimed unit has lost over 80% of its initial strength.

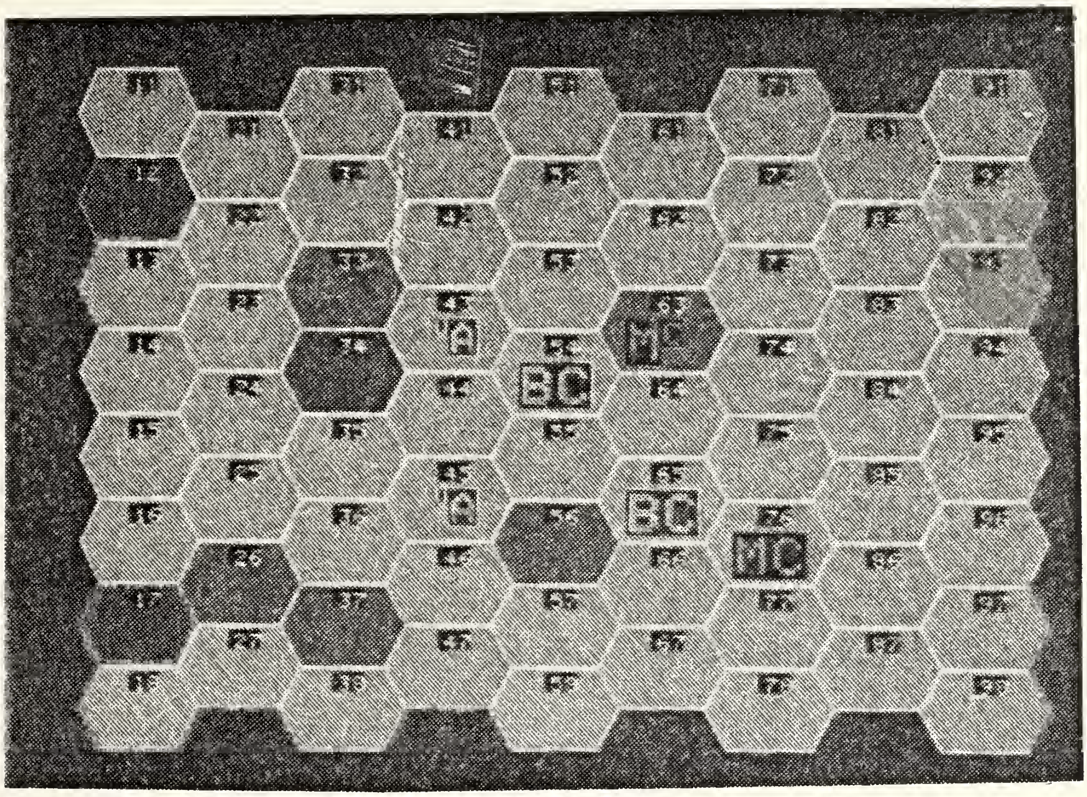


Figure 2.1 The Playing Board Layout.

The blue letter A is the only exception. At the beginning of the simulation, there are no Blue sensor-aimed units. See Chapter IV for an explanation of why there are no Blue sensor-aimed units, and how to transform Blue unaimed to Blue sensor-aimed units. As the conflict progresses and Blue unaimed units are transformed into Blue sensor-aimed units, the size of the blue letter A increases. The letter A becomes the next larger size when the Blue sensor-aimed unit gains 20% of the Blue unaimed units' initial strength. Also note, that as the Red units fire on the Blue sensor-aimed units, the letter A decreases in size because of attrition.

D. SEQUENCE OF PLAY

The simulation is made up of four phases each of which the user is an integral part. The four phases are

(1). The user input

(2). Selection of the Red Forces' route of attack

(3). The conflict simulation

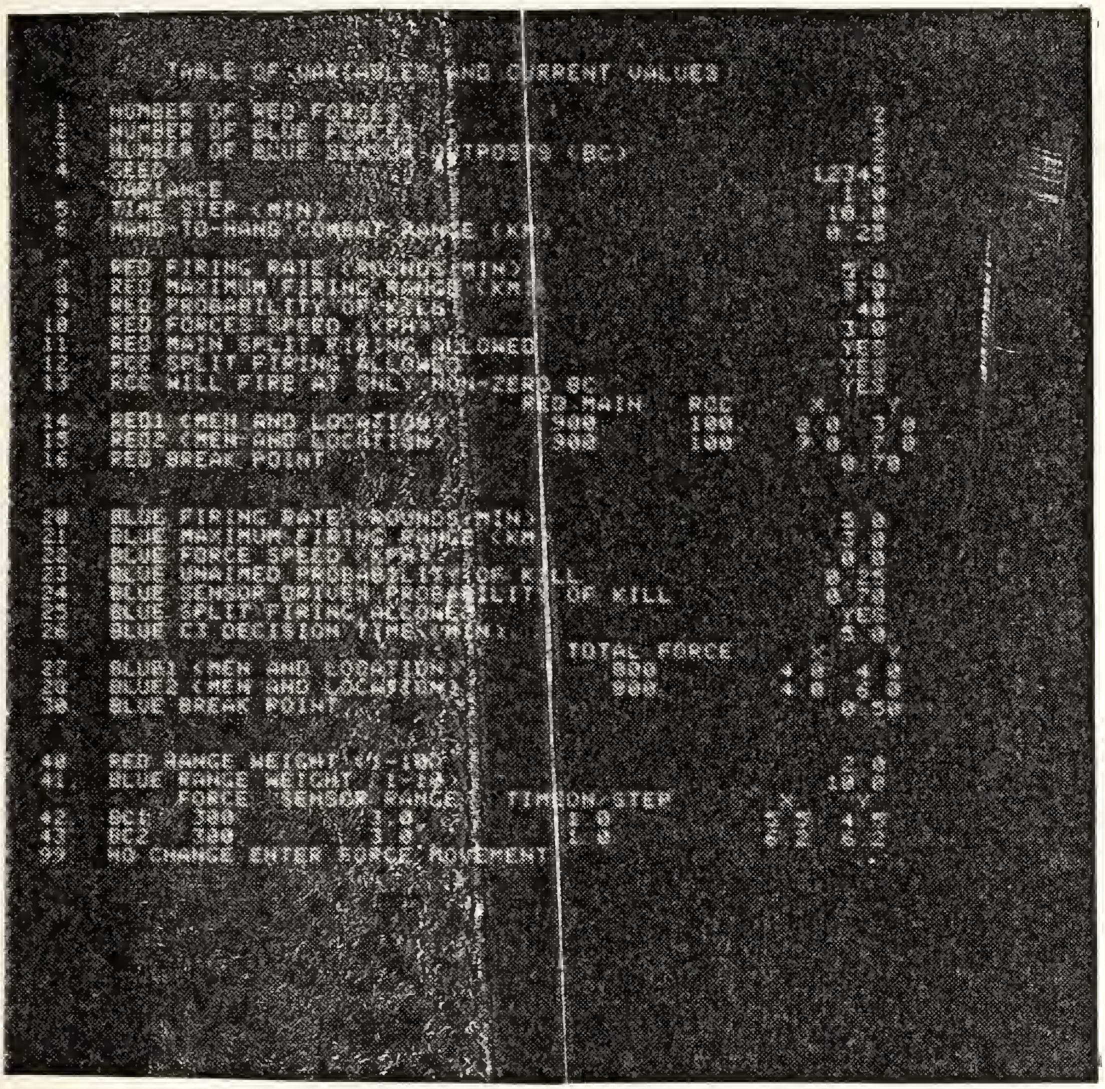
(4). After-battle analysis

1. The User Input

The first step in simulating the conflict is to initialize all of the program's variables. To do this the

user is presented a menu on the RAMTEK graphics display titled 'Table of Variables and Current Values' (see Figure 2.2) in which all the variables have been initialized with default values. During this phase, the user can change any variable's initial value. To change a value of a variable, or to obtain an explanation of the meaning of a variable, the user enters the variable's line number on the RAMTEK's keyboard and presses the return key. For example, if the user desires to change the hand-to-hand combat range, he enters the number 6 and presses the return key. The explanation of the variable hand-to-hand combat range is then displayed on the screen (see Figure 2.3). The user may change the hand-to-hand combat range by entering a new range, or he can elect to leave the range at its current value by simply pressing the return key without pressing any other key.

At the beginning of the simulation, all the variables have initial values. If the user changes the value of a variable, then that value becomes the default value for all the remaining simulation runs for the session. This capability allows the user to incrementally change a variable, determine its impact on the outcome of the battle, and



```
        TABLE OF VARIABLES AND CURRENT VALUES
 1   NUMBER OF RED FORCES                                   2
 2   NUMBER OF BLUE FORCES                                  2
 3   NUMBER OF BLUE SENSOR OUTPOSTS (BC)                    2
 4   SEED                                               12345
     VARIANCE                                             1.0
 5   TIME STEP (MIN)                                     10.0
 6   HAND-TO-HAND COMBAT RANGE (KM)                      0.25
 7   RED FIRING RATE (ROUNDS/MIN)                         7.0
 8   RED MAXIMUM FIRING RANGE (KM)                [illegible]
 9   RED PROBABILITY OF KILL                      [illegible]
10   RED FORCES SPEED (KPH)                               3.0
11   RED MAIN SPLIT FIRING ALLOWED                        YES
12   ROC SPLIT FIRING ALLOWED                             YES
13   ROC WILL FIRE AT ONLY NON-ZERO BC                    YES
                            RED MAIN    ROC        X     Y
14   RED1 (MEN AND LOCATION)   900      100       0.0   3.0
15   RED2 (MEN AND LOCATION)   900      100       0.0   7.0
16   RED BREAK POINT                                     0.70

20   BLUE FIRING RATE (ROUNDS/MIN)                        3.0
21   BLUE MAXIMUM FIRING RANGE (KM)               [illegible]
22   BLUE FORCE SPEED (KPH)                               0.0
23   BLUE UNAIMED PROBABILITY OF KILL             [illegible]
24   BLUE SENSOR DRIVEN PROBABILITY OF KILL       [illegible]
25   BLUE SPLIT FIRING ALLOWED                            YES
26   BLUE C3 DECISION TIME (MIN)                          5.0
                              TOTAL FORCE          X     Y
27   BLUE1 (MEN AND LOCATION)     900             4.0   4.0
28   BLUE2 (MEN AND LOCATION)     900             4.0   6.0
30   BLUE BREAK POINT                                    0.50

40   RED RANGE WEIGHT (1-10)                              2.0
41   BLUE RANGE WEIGHT (1-10)                            10.0
          FORCE  SENSOR RANGE  TIME ON STEP        X     Y
42   BC1   300       1.0          1.0       [illegible] 4.5
43   BC2   300       1.0          1.0       [illegible] 6.5
99   NO CHANGE ENTER FORCE MOVEMENT
```

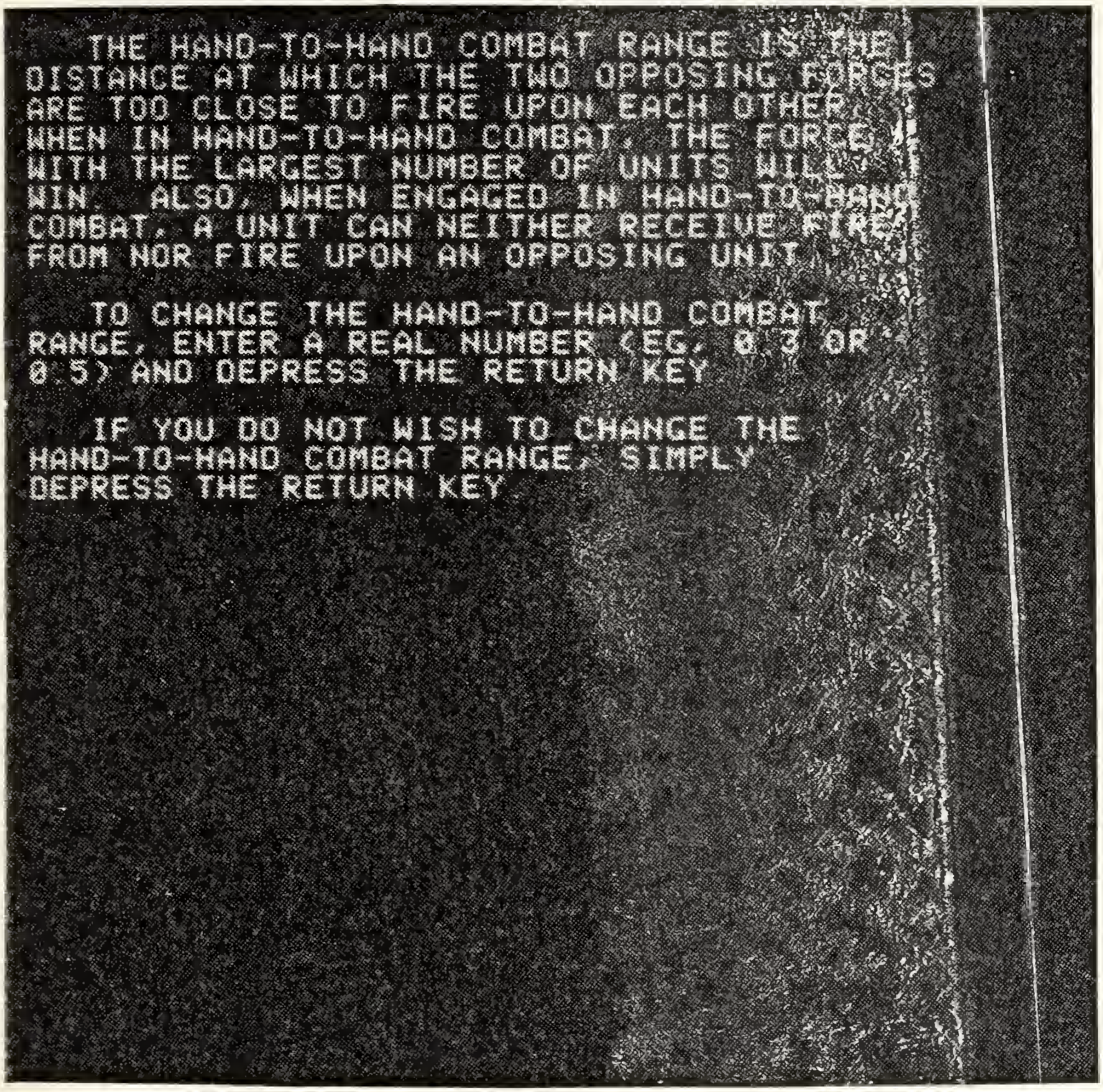
Figure 2.2   Table of Variables and Current Values.

then change the value of another variable and not have to re-enter all the previous changes which the user had made.

After all the variables have been initialized, the user enters 99 and the second phase of the simulation is entered.

2. Selection of the Red Forces' Route of Attack

The second phase of the simulation is determining the Red force's route of attack. In the present version of the simulation, only the Red force is allowed to move. Therefore, the user must plan a Red route of attack and enter it in the computer. This is simplified for the user through the use of the trackball. The user designates all turning points by moving the trackball cursor to the hex where the turning point is to occur and depressing the enter key on the trackball. (The Red force travels in a straight line from turning point to turning point.) When the user has entered the desired path of the Red force, the cursor is then moved into the red area at the bottom of the screen and the enter key on the trackball depressed. (See Figure 2.4) At this time, the program automatically finds the location of the closest Blue Main force and ensures the Red unit will move from the final turning point to within hand-to-hand



```
   THE HAND-TO-HAND COMBAT RANGE IS THE
DISTANCE AT WHICH THE TWO OPPOSING FORCES
ARE TOO CLOSE TO FIRE UPON EACH OTHER.
WHEN IN HAND-TO-HAND COMBAT, THE FORCE
WITH THE LARGEST NUMBER OF UNITS WILL
WIN.  ALSO, WHEN ENGAGED IN HAND-TO-HAND
COMBAT, A UNIT CAN NEITHER RECEIVE FIRE
FROM NOR FIRE UPON AN OPPOSING UNIT.

   TO CHANGE THE HAND-TO-HAND COMBAT
RANGE, ENTER A REAL NUMBER (EG, 0.3 OR
0.5) AND DEPRESS THE RETURN KEY.

   IF YOU DO NOT WISH TO CHANGE THE
HAND-TO-HAND COMBAT RANGE, SIMPLY
DEPRESS THE RETURN KEY.
```

Figure 2.3   Hand-to-Hand Combat Range Explanation.

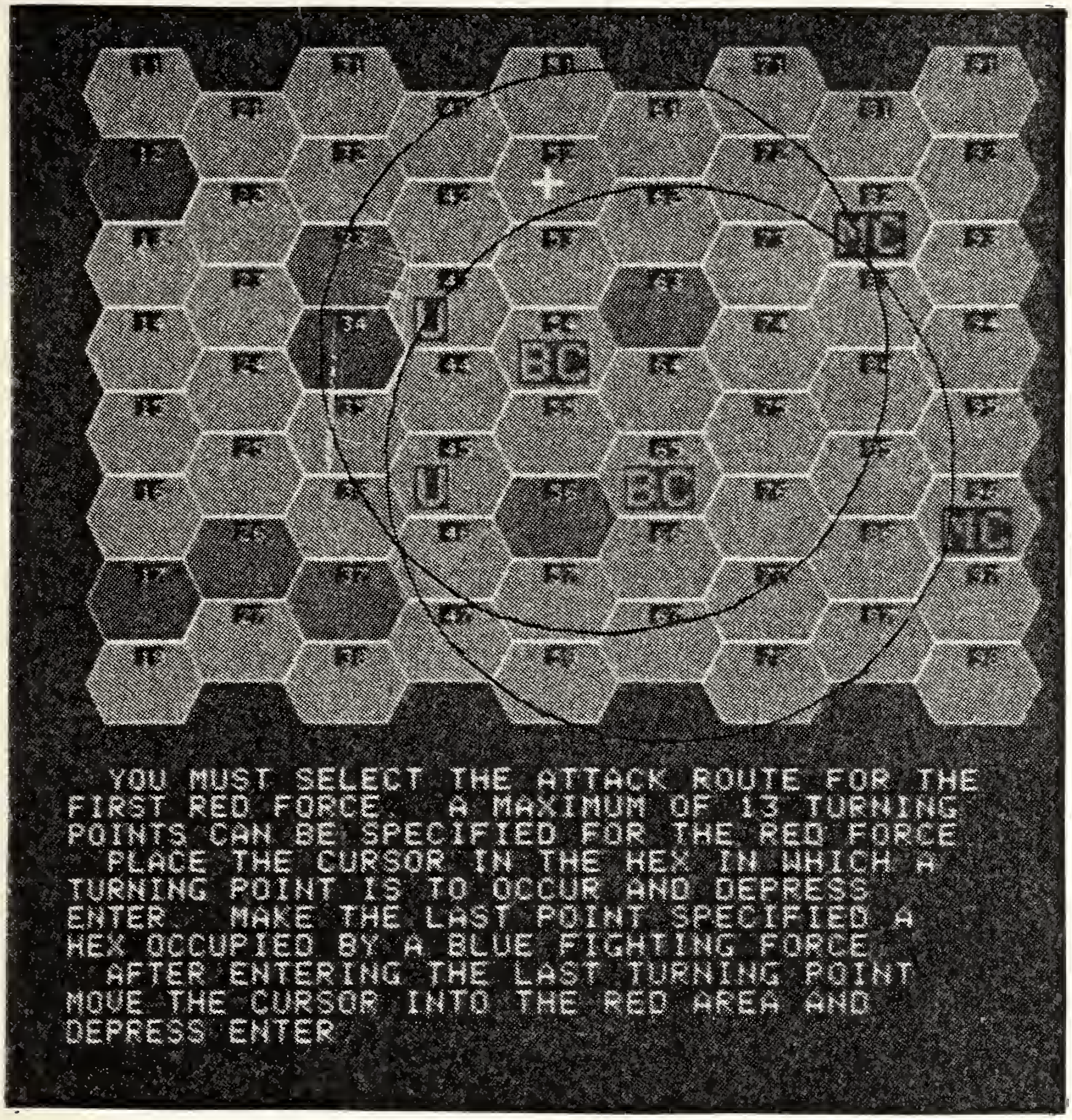
combat range of the Blue Main force. (This feature ensures the battle will terminate.)

Figure 2.4 shows two black circles drawn on the playing board with a BC outpost at each circle's center. The circle represents the approximate sensor range for the BC outpost at its center. If the Red force enters the circle during its attack, the BC outpost will sense the presence of the Red force and this will allow some of the Blue unaimed (relatively ineffective) units to be transformed into Blue sensor-aimed (relatively effective) units. Therefore, the Red force should attempt to stay outside of the BC sensor range or to travel on the fringe of the black circle to lessen the effect of the BC on the outcome of the conflict.

Once the route of attack has been selected and entered into the computer, the battle simulation (phase three) begins.

3. Phase Three

During phase three of the simulation, the Red units attack and the Blue units retaliate. As the battle progresses, the computer-generated color displays of the battlefield show the letters representing the different



YOU MUST SELECT THE ATTACK ROUTE FOR THE FIRST RED FORCE. A MAXIMUM OF 13 TURNING POINTS CAN BE SPECIFIED FOR THE RED FORCE. PLACE THE CURSOR IN THE HEX IN WHICH A TURNING POINT IS TO OCCUR AND DEPRESS ENTER. MAKE THE LAST POINT SPECIFIED A HEX OCCUPIED BY A BLUE FIGHTING FORCE. AFTER ENTERING THE LAST TURNING POINT MOVE THE CURSOR INTO THE RED AREA AND DEPRESS ENTER

Figure 2.4 User Inputting the Red Force Route of Attack.

units reducing in size as the units are depleted. Also, it shows the Red unit's letters moving across the playing board.

In this phase, the user has an opportunity to examine four status reports. The Red and Blue Force Status reports (Figures 2.5 and 2.6) give the exact location of each force and the strengths for each unit type as well as the total strength for each force. Figure 2.7 shows the Blue Sensor Outpost Status report which provides the user with the number of assets remaining at each BC outpost. Figure 2.8 displays the ranges between all the forces involved in the conflict. The user may examine any status report by placing the trackball cursor over the letters of the status report he wants to examine and depressing the enter key. For example, the user could examine the Blue Force Status report by placing the cursor over the word "BLUE" at the bottom of the screen and depressing the enter key. When the user places the cursor over the letters "MODEL", the next game step of the simulation is displayed.

The displays provided during this phase of the simulation allow the user to visualize what is taking place on the battlefield. For example, the user can determine if a

unit is being fired upon by more than one opposing unit or if a unit is dividing its firepower among all targets of opportunity. The battle displays provide the user with pictorial representations of the battle instead of columns of numbers in a printout.

When the force strengths of all the Red units, or all the Blue fighting units, are equal to zero (ie, one force has won the battle by annihilating the other) then the user enters phase four of the simulation by placing the cursor over the letters "MODEL" and depressing the enter key.

4. After-Battle Analysis

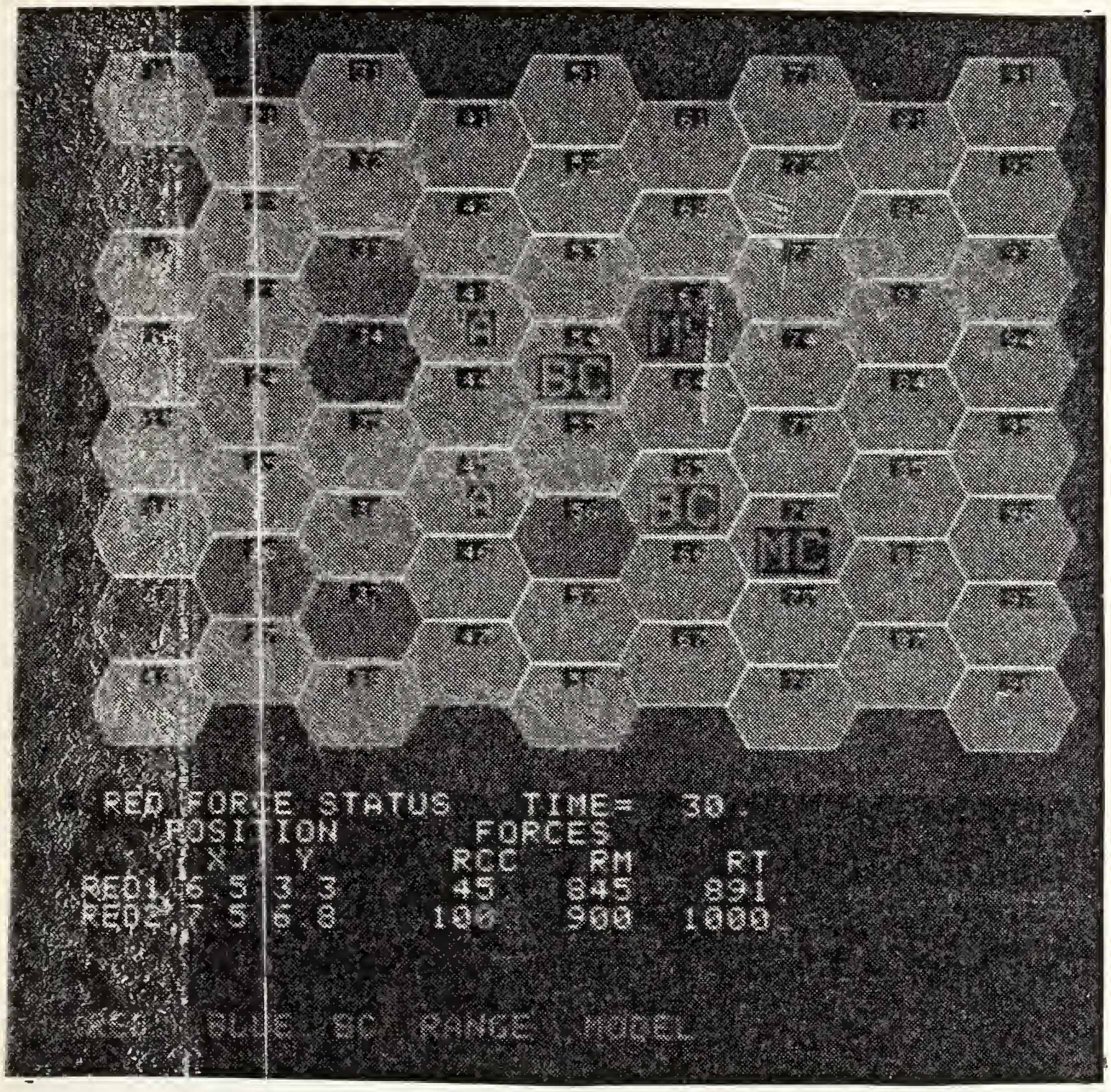
The final phase of the simulation is the Force vs Time graph. (See Figures 2.9 and 2.10) This graph allows the user to plot the force strength vs time of all the units involved in the battle. It also plots the break points (a user input) on the total force strength line (RT1, RT2, BT1, BT2, and BT3). A total force strength line is the sum of the units which form the force. For example, RT1 is the sum of RM1 and RCC1, while BT1 is the sum of BU1 and BA1. This allows the user to hypothesize that the outcome would have been different had the units retreated at their respective break points.

Figures 2.9 and 2.10 are examples of graphs the user can examine. The plotted lines are color coded and have letters spaced along them to aid the user in determining which line is associated with which force. The Red forces all have red lines, the Blue Main forces have blue lines and the BC outposts have black lines. The letter associated with each line is shown in the blue menu area at the bottom of the figure. For example, the Red force's total strength line, RT1, has the red letter A spaced along it, while the Blue force's total strength line, BM1, has the blue letter G spaced along it.

It should be noted by the user that there is a maximum of 50 data points per line that can be plotted by this routine. Therefore, if a battle simulation has more than 50 game steps, only the first 50 data points will be plotted.

The user selects the unit to plot by placing the trackball cursor over the letters representing the unit and depressing the enter key. If the user wishes to clear the screen and replot the forces, then place the cursor over the words "NEW PLOT" and depress the enter key. To terminate this phase of the simulation, the user places the cursor over the letters "RETURN" and depresses the enter key.

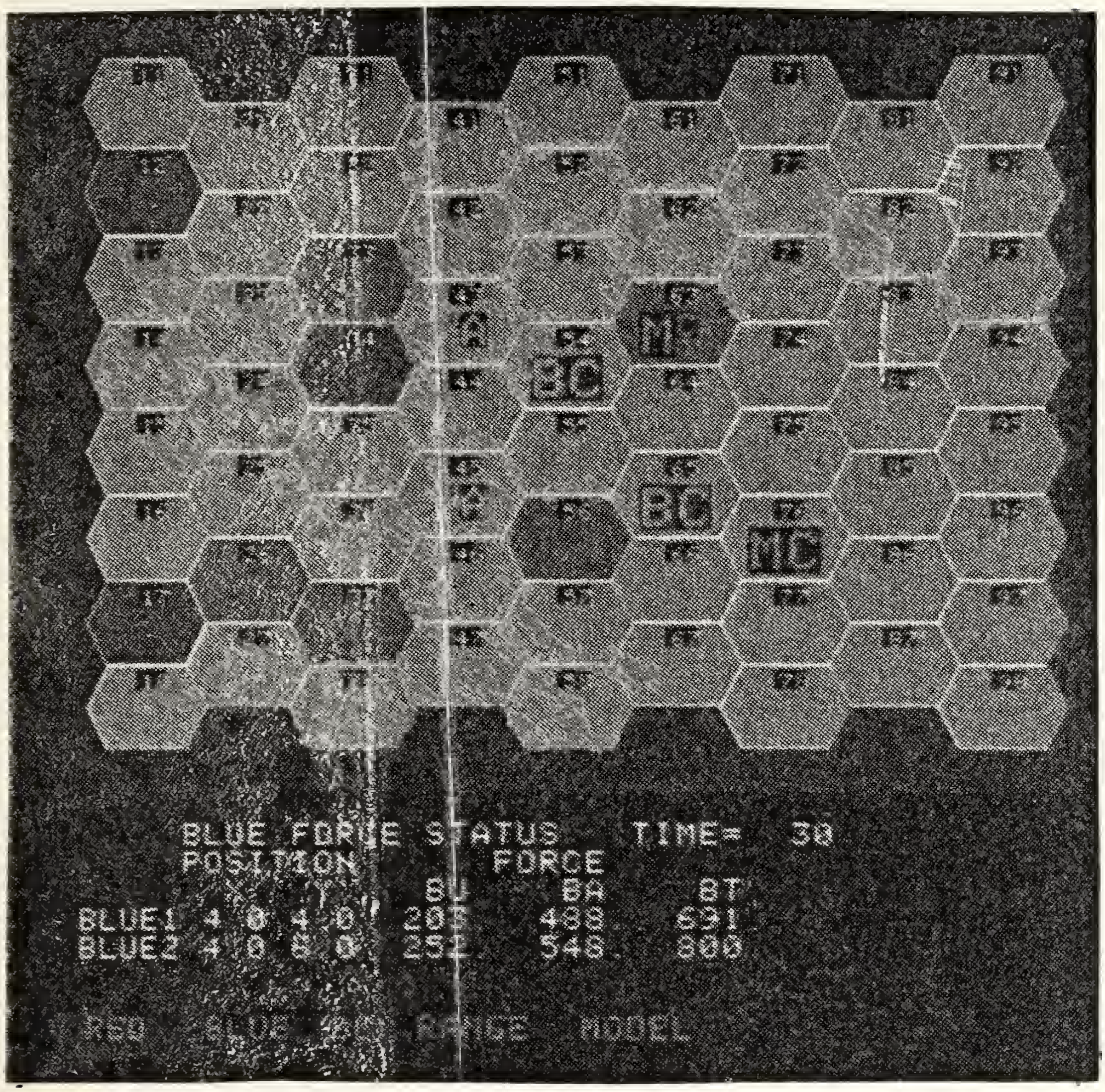
At this point in the simulation the user can either return to phase one and start another simulation, or exit the program. If the user returns to phase one, all the variables are initialized to the values of the last simulation run and the user may again change any value of any variable. Also, the user may use the same route of attack as the previous simulation, or may enter a new attack route.



```
RED FORCE STATUS    TIME=  30.
   POSITION        FORCES
     X    Y      RCC    RM    RT
RED1 6.5  3.3     45   845   891
RED2 7.5  6.8    100   900  1000

RED  BLUE  BC  RANGE  MODEL
```

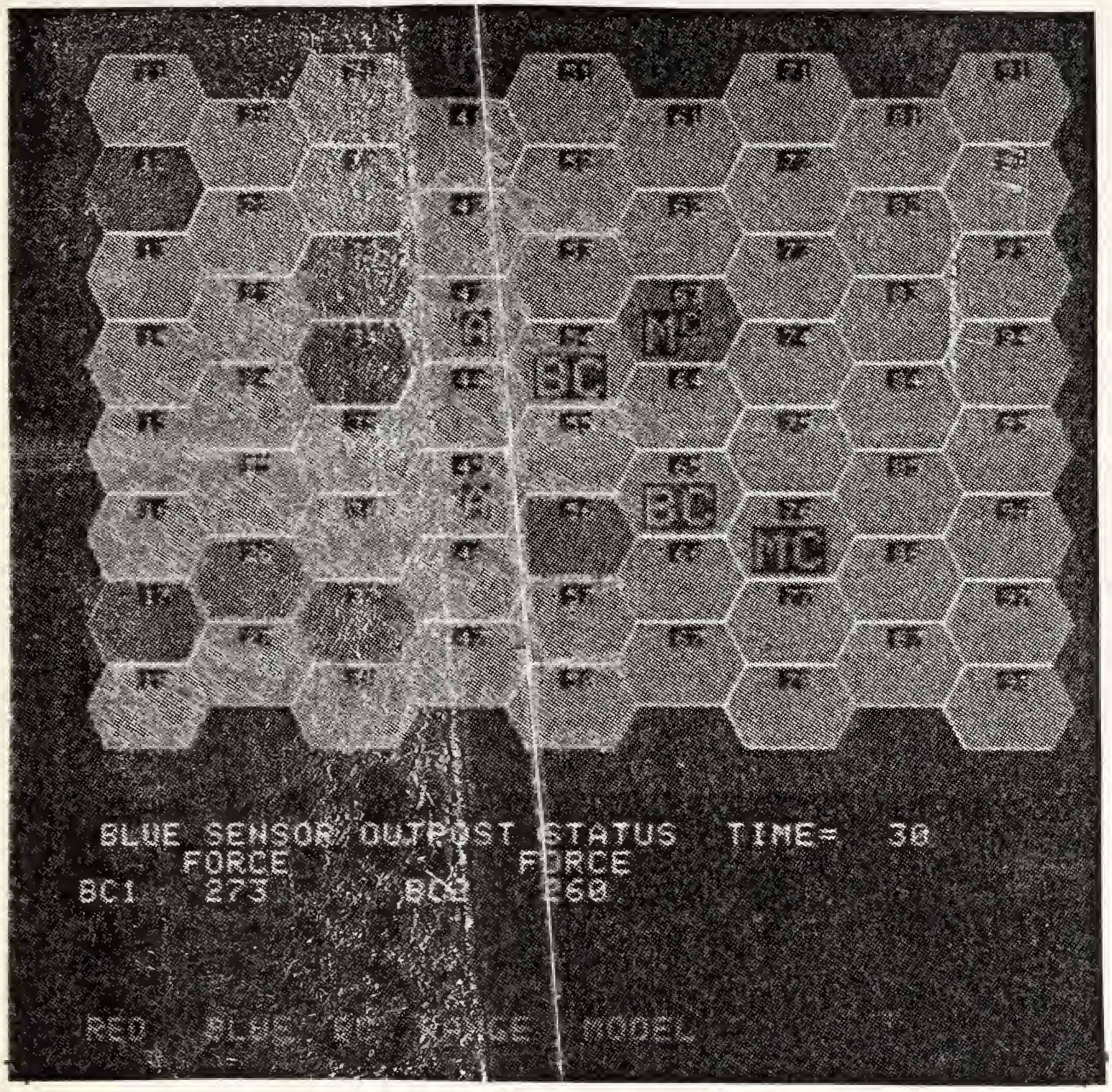
Figure 2.5   Combat Simulation Showing Red Force Status.



```
    BLUE FORCE STATUS    TIME=  30
    POSITION        FORCE
     X   Y     BU      BA     BT
BLUE1 4 0 4 0  203     489    691
BLUE2 4 0 8 0  252     548    800

RSG  BLUE [illegible] RANGE  MODEL
```

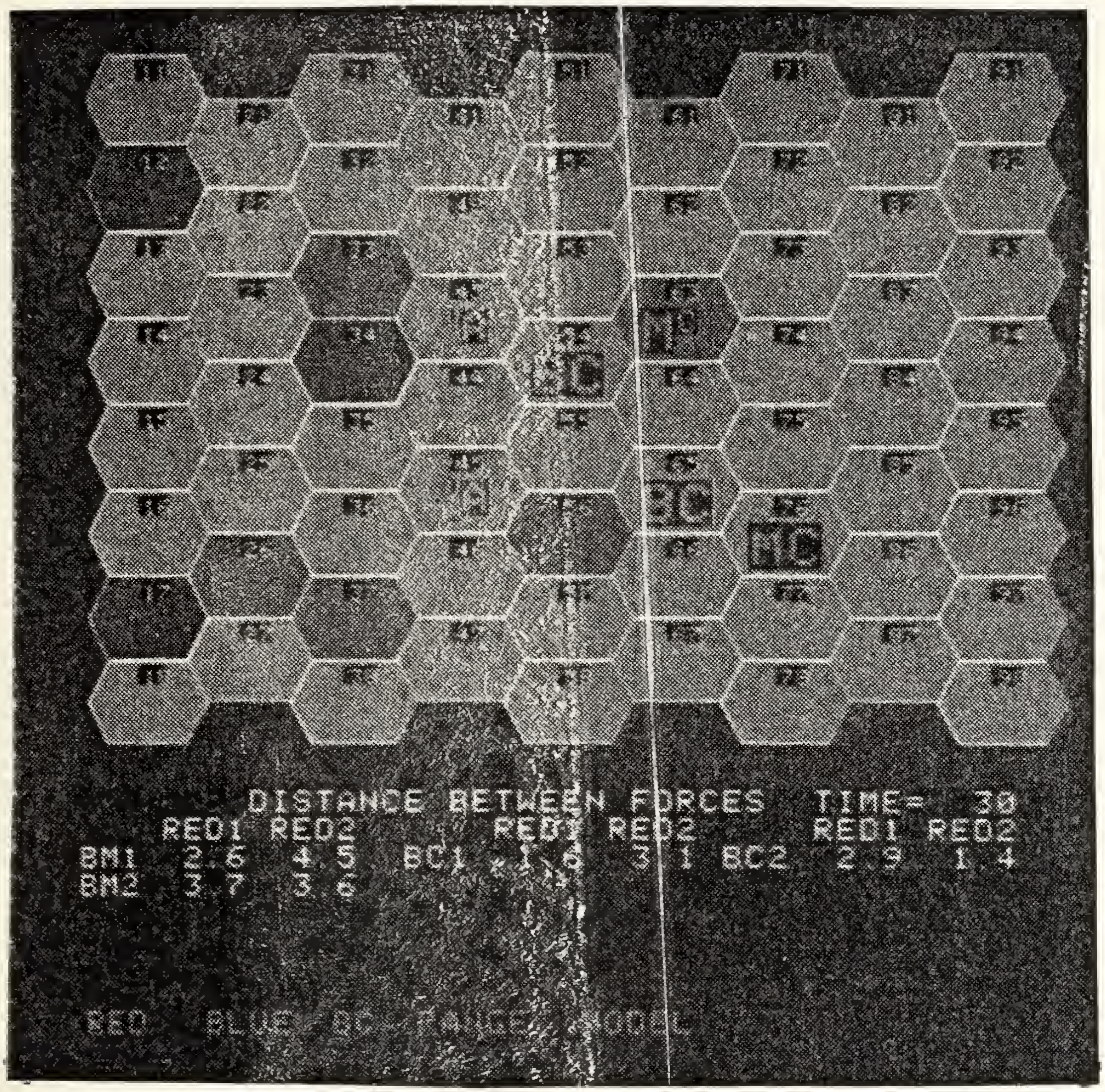
Figure 2.6 Combat Simulation Showing Blue Force Status.



```
 BLUE SENSOR/OUTPOST STATUS  TIME=  30
      FORCE              FORCE
BC1   273         BC2    260

RED  BLUE [illegible] FORCE MODEL
```

Figure 2.7 Combat Simulation Showing BC Outpost Status.



```
          DISTANCE BETWEEN FORCES  TIME=  30
     RED1 RED2          RED1 RED2         RED1 RED2
BM1  2.6  4.5  BC1  1.6  3.1 BC2  2.9  1.4
BM2  3.7  3.6
```

Figure 2.8 Combat Simulation Showing Range Between Forces.

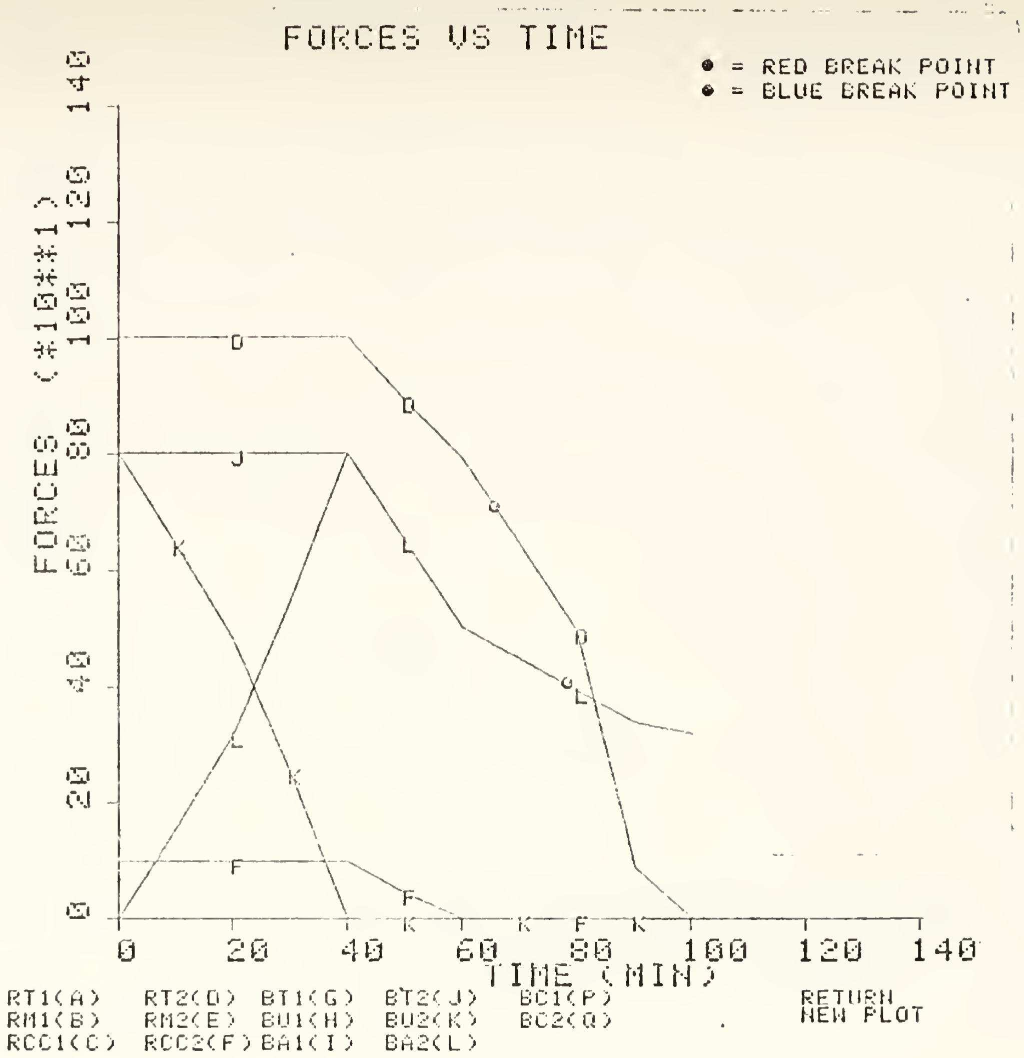


Figure 2.9 Example 1 of a Force Strength vs Time Plot.

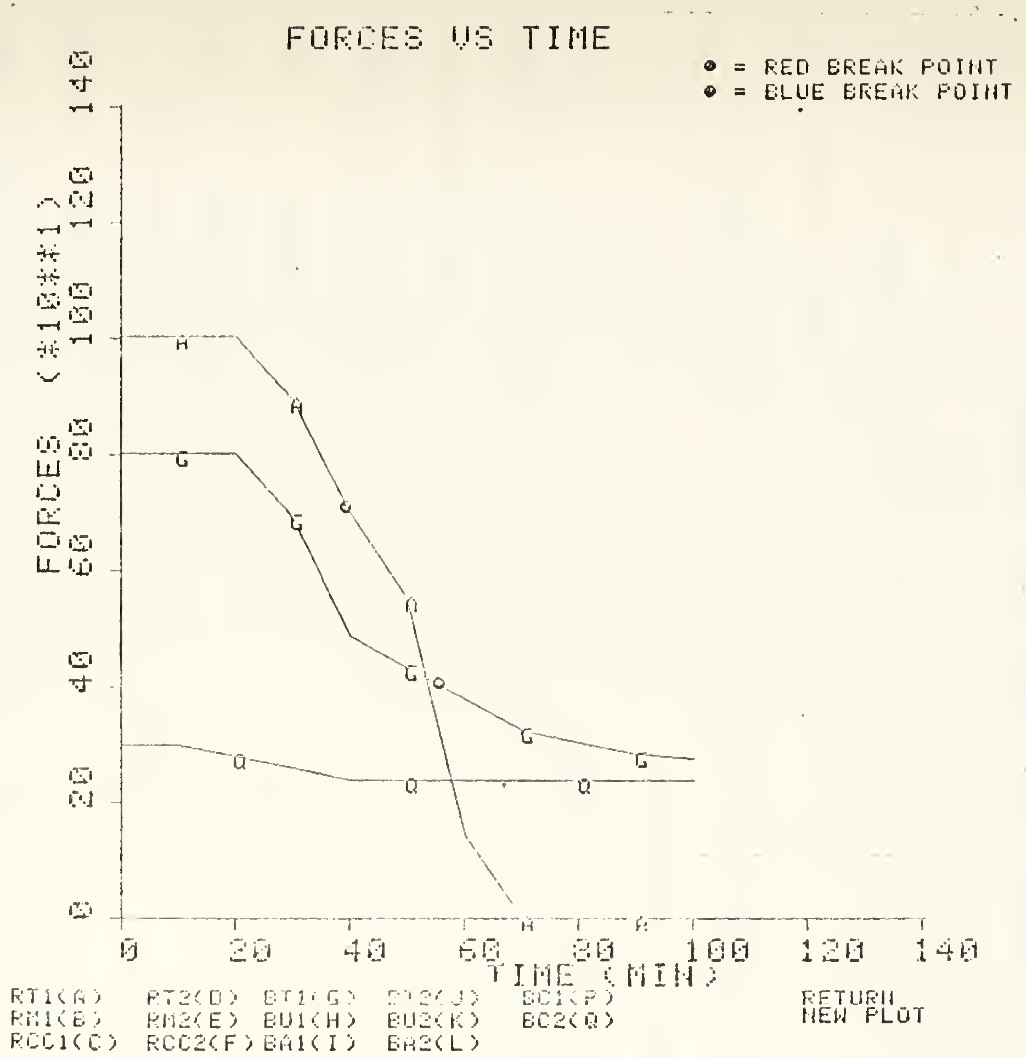


Figure 2.10 Example 2 of a Force Strength vs Time Plot.

III. A SIMPLE C2 MODEL

This section of the paper will provide the user with a very basic explanation of a C2 model and describe the functions of the C2 process. The simulation attempts to incorporate all the functions of the C2 process and it also tries to capture the time dependence of the C2 information in a combat situation.

The primary goal of any C2 system is to control a hostile environment. To control the environment means either to maintain the status quo between forces, or to obtain a more advantageous position, or to bring about a successful conclusion to a confrontation or battle. In this computer simulation, the goal of the Blue C2 is to bring about a successful conclusion of a battle with the smallest loss of assets possible.

The C2 process is depicted in Figure 3.1. For a C2 system to be effective in controlling its environment, it must first be able to sense its surroundings and then process the sensed data and compare these results with the desired state of the environoment. After the comparison is made, a C2 system must be able to decide on courses of

action tending to make the sensed state of the environment and the desired state of the environment the same. Finally, a C2 system must choose a course of action for its forces to follow to achieve it's goal, ie, to bring the sensed state of the environment close to the desired state of the environment.

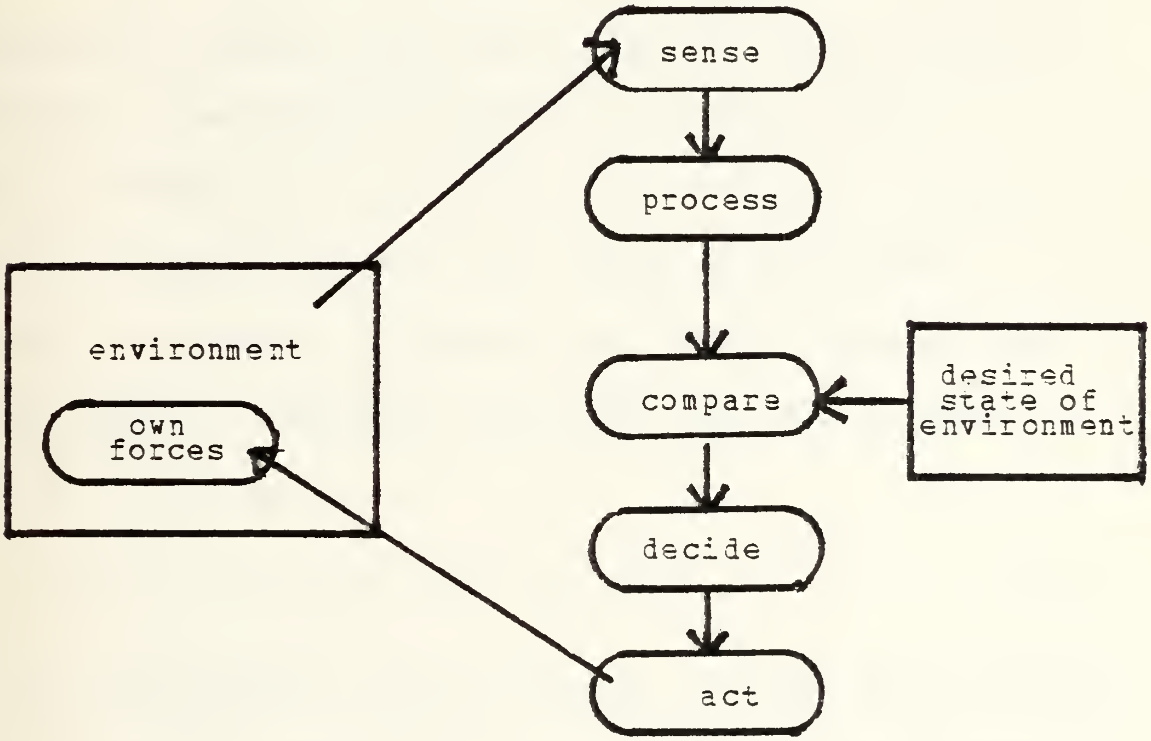


Figure 3.1 The C2 Process.

Not explicitly shown in Figure 3.1, but an integral part of the C2 system, is the time component for each function of the C2 process. If the C2 system can not bring its forces

to bear on the environment in a timely manner, then the battle may be lost before its forces have a chance to act.

The times associated with each of the functions of the C2 process are shown in Figure 3.2. Ts is the time required to sense the initiated hostile action. It will next take a time, Tp, to process the sensed data and to pass the processed data to the comparision function where time Tc is required to evaluate the data and pass it to the decision function. The decision function will need a time Td to reach a decision and time, Ta, will be taken to prepare a course of action and pass this choice to the forces. Finally, the forces will need time, Tf, to implement the orders given. Therefore, the total time, Tt, involved in the C2 process is

$$Tt = Ts + Tp + Tc + Td + Ta + Tf \qquad (3.1)$$

If one assumes a hostile action is initiated at time T1 and at time T2 the hostile action will stop the C2 system from achieving its goal, then the sum of the times Tt associated with each function of the C2 process should be less than T2-T1 if the C2 system is to be successful.

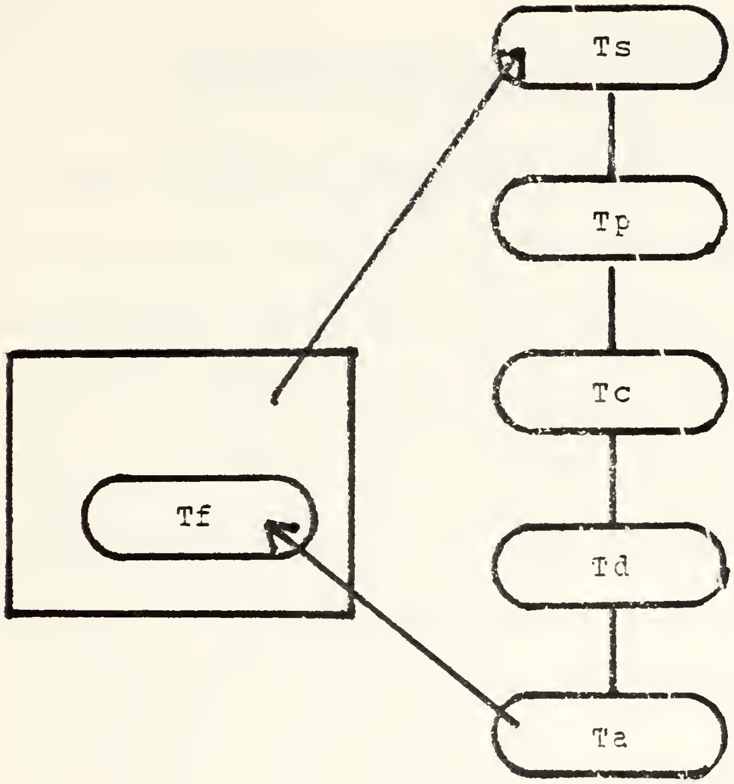


Figure 3.2 The Time Components of the C2 Process.

Figure 3.2 The Time Component [illegible]

IV. IMPLEMENTATION OF THE C2 MODEL

The previous section illustrated a very general and simple C2 system, and pointed out that time plays an important factor in the C2 process. For a computer program to simulate the C2 process, it must represent both the C2 functions and the time factor. This section will describe how the simulation presented in this thesis represents the C2 process and some of the decisions which must be made by the user. It is important to note that a particular combat scenario is described here for illutration. The basic ideas of C2-affected combat, and the graphical display can, of course, be extended to describe a variety of other interesting situations.

A. THE OPPOSING FORCES

In the present specific scenario, there are two types of opposing forces, Red forces and Blue forces. The Red force attacks the Blue force with the goal of annihilating the Blue fighting force to gain control of the territory occupied by Blue. The Blue force, on the other hand, is attempting to stop the Red force's advance into Blue's terrain.

The Blue force has the option of putting some of its resources into a C2 system. In the present scenario, the Blue C2 force converts "unaimed" (relatively ineffective) Blue forces into "aimed" (relatively effective) Blue forces; the latter wreaks greater attrition upon Red forces than the former. The Blue C2 system could give the Blue force a decisive edge during a battle depending on the decisions made by the Red force and how well the C2 system operates. Presently, the Blue force has no counter-command and control (C-C2) resources.

The Red force, in the current simulation, has no explicit C2 assets but it does have at its disposal C-C2 resources. These C-C2 resources can negate the C2 benefits of the Blue force.

B. SIMULATING THE C2 PROCESS

The Blue forces are composed of three types of units

1). Blue command control sensor outposts (BC)
2). Blue unaimed units (BU) and
3). Blue sensor-aimed units (BA)

The first three function of the C2 process (ie, sensing, processing, and comparing) are performed by the Blue command and control sensor outpost (BC). The BC outposts have no

fighting capability. They do have, however, sensors which can detect the approaching Red units. The BC sensors, as is true of all sensors, have a maximum detection range beyond which the Red units are undetectable. When the BC outposts do detect the Red units approaching, they process the data and send information to the Blue Main force (BM) which allows Blue unaimed units (BU) to be transformed into Blue sensor-aimed units (BA). The BA units are then brought to bear upon the advancing Red units.

An expression was developed which gives the maximum number of BU units to be transformed to BA units during each game step by each BC outpost for each Blue fighting force. It expresses the maximum number of units as directly proportional to the amount of resources allocated to the BC outpost and, directly proportional to the fraction of time per game step that the BC outpost is turned on, ie, actively searching for approaching Red units. It also expresses the maximum number of units as being inversely proportional to the distance the BC outpost is from the approaching Red unit it is reporting on (the larger the distance, the less accurate the targetting information) and, inversely proportional to the distance the BC outpost is from the Blue fighting force it is sending its report to.

The maximum number of BU units which can be transformed into BA units at each Blue Main force by a single BC outpost in a game step is calculated using the following formula:

$$BAMAX = \frac{TIMEON * BC}{1 + \frac{RNGBC}{RDWT} + \frac{RNGBLU}{BWT}} \qquad (4.1)$$

where

BAMAX = maximum number of BU units that can be changed to BA units during a game step;

TIMEON = the fraction of a game step during which the BC sensor is turned on;

BC = the amount of resources a BC unit has available during a game step;

RNGBC = the distance the Red force is from the reporting BC unit;

RNGBLU = the distance the Blue force is from the reporting BC unit;

RDWT = the importance the user places on the distance the Red force is from the BC unit;

BWT = the importance the user places on the distance between the Blue force receiving the report and the reporting BC unit.

The value of the variables TIMEON, BC, RDWT, and BWT are user supplied during the input phase of the simulation discussed in Chapter II.

RDWT and BWT weight factors can be assigned a value between 1.0 and 10.0. A weight of 1.0 means the range between the BC outpost and the force (either Red or Blue) is the most important factor in the transformation of BU units. A weight of 10.0 signifies that the range has little impact on the number of BU units which could be transformed. It should be noted that once all the BU units have been transformed to BA units, the BC outposts do nothing further to aid the BM forces.

In the present simulation, the BC outposts are considered to be networked together. Therefore, if a Red force enters the BC sensor range, the transformation of BU units to BA units occurs for all Blue Main forces. During the user input phase, the variable BWT is entered allowing the use to regulate the transformation rate at the Blue Main units based on data received from distant BC outposts. If a user places the BC outpost a large distance from a Blue Main force and determines that BWT is very important (ie, 1.0 or 2.0), then fewer BU units would be transformed to BA units

than if BWT was insignificant. The user should make a trade-off between early detection of approaching enemy forces and the number of BU units transformed to BA units in each game step.

Similarly, altering the variable RDWT allows the user to control the tranformation rate of BU units to BA units based on the distance the advancing Red force is from the BC outpost. Increasing the importance of the range from the Red force to BC outpost means that the number of BU units transformed during a game step would be less if the Red force was on the fringe of the BC sensor range than if the Red force was directly on top of the BC outpost.

The decision and action process of the C2 system are performed at the Blue Main (BM) force. The BM force is composed of both BU and BA units. The mission of the BM force is to convert BU units to BA units. The goal of the Blue force is to have as many BA units as possible when fighting because the BA units have a much higher probability of kill than do the BU units. However, both the BA unit's probability of kill and the BU unit's probability of kill are user supplied values. Therefore, the effect of the BA units on the battle outcome depends on the initial values of the probability of kill.

The number of BU units which are actually changed to BA units at each Blue Main force is calculated using the following equation:

$$BUCVT = BAMAX*(1 - \exp(-STEP/(NBC*BTIME))) \tag{4.2}$$

where

BUCVT = the number of BU units converted to BA units at a Blue Main force

BAMAX = the sum of equation (4.1) for all BC units reporting to a Blue Main force. (There can be up to 6 BC units operating at a time)

NBC = number of active BC units during the game step (If BC = 0, then BUCVT = 0)

BTIME = time required by the Blue force to act on the hostile action taken by the Red force

STEP = number of minutes in a game step

The value of the variables NBC, BTIME, and STEP are user supplied during the input phase of the simulation. (The user should be aware that equation (4.2) is valid only for a range of NBC between 0 and 6.)

In Chapter III it was stated that time was a very important factor of a C2 system. Time is introduced into this

simulation through the variable BTIME. BTIME represents the time required for the Blue force commander to convert his unaimed units to sensor-aimed units. BTIME is the sum of the time (ie, Tt of equation (3.1)) for each function of the C2 process. The larger the value of BTIME, the smaller the number of BU units which are transformed to BA units during a time step.

Equation (4.2) also incorporates another important aspect of command and control into the simulation. That aspect is that more command and control is not necessarily better than less command and control. The variable NBC in equation (4.2) (the number of active, reporting BC outpost during a game step) can greatly influence the actual number of BU units converted to BA units at each Blue Main force. It is assumed that as more BC outposts report to a BM headquarters, it will take more time for the C2 system to process the reports, decide on the required action and communicate that course of action to the forces in the field. Therefore, as more BC outposts are added to the C2 system, the system efficiency (ability to transform units) will decrease. (It should be noted that if Blue adds more BC outposts and EAMAX (from equation (4.1)) does not

increase, then this is a waste of valuable resources. Also, the Blue force commander must still process whatever information the additional BC outposts are sending to him, even if it does not help his fighting forces.)

One way to make the C2 system more efficient is to lessen the amount of decision time (BTIME) required by the Blue force commander. If, for each additional BC outpost added to a C2 system, the time required to react to a hostile action is reduced, then the number of units transformed in equation (4.2) would increase. This could be accomplished if the user were to stipulate there was an information filter being used by the Blue force commander. Of course, the resources devoted to filtering information should be at the expense of fighting units, ie, fewer BU units at the start of the battle.

A user has to make a trade-off between the C2 system efficiency and ensuring the battlefield is covered by his sensors. If all avenues of approach by the Red force are covered by BC sensors, then the Blue will at least have some BA units when Red comes within firing range. However, if Blue does not cover the battlefield with his sensors, then Red could conceivably attack Blue when Blue only has BU units and thereby easily defeat the Blue force.

C. THE HOSTILE FORCES

In this simulation the Red units are the aggressor force. The Red force is composed of two types of units, Red Main (RM) and Red counter-command and control (RCC). The Red Main units only attack the BM forces, ie, BU and BA units. The RCC units attack the BC outposts when they are active and within range. If there are no active, in-range BC outposts, the RCC will join the RM force in attacking the BM force.

The RCC units are designed to give the Red force an opportunity to counter the C2 system of the Blue force. They allow the Red force to prevent the Blue force from converting unaimed units to sensor-aimed units by reducing the amount of BC resources (BC in equation (4.1)) available at a BC outpost.

D. PRE-BATTLE DECISIONS

Several command and control (C2) decisions must be made by both the Red and Blue forces prior to the start of the conflict.

The Blue force C2 decisions include the following factors:

1. The number of Blue C2 sensor outposts that will be deployed (0 to 6).

2. The placement of the BC outposts. The Blue force should ensure that as much as possible of the battlefield is covered by sensors after considering loss of processing efficiency due to more BC outpost.

3. The quantity of BC resources allocated to each BC outpost. This decision directly affects the number of units that can be transformed per unit time; it also affects the number of fighting units the Blue force starts the battle with. The Blue fighting force typically begins with given resources, and then, as assets are placed in the BC outposts, equivalent resources should be deducted from the Blue fighting force. Allocation of resources between C2 and main fighting forces is a crucial decision for Blue.

4. The fraction of a game step during which each BC outpost will be turned on. This decision variable affects both the maximum number of BU units which can be converted, and the attrition rate for the BC outpost. The longer the outpost is turned on or

active the easier it is identified and targetted by Red C-C2.

5. The range of each BC outpost sensor.

6. The time (BTIME) required by the Blue force to react to the hostile action of the Red force. (Many of the trade-offs and issues discussed earlier should be take into account by the user when assigning a value to this variable.)

7. The importance the user places on the distance the BC outposts are from both the Red units (RDWT) and the Blue fighting units (BWT).

8. The placement of the Blue fighting units. (This decision should be made in conjunction with deciding a value for BWT.)

9. Blue firing doctrine: whether the BM units will fire at only the closest Red unit or will split its firing equally among all the targets within firing range.

The Red C2 decisions include the following factors:

1. The number of Red units which will be designated as RCC units.

2. Whether or not the RCC will fire on only the closest BC outpost, or will divide its fire power equally among all BC outposts within firing range.

3. The Red counter-command and control must also decide in advance whether or not it will know if a BC outpost's resources have been reduced to zero. If the RCC will not know a BC outpost has been eliminated, then it should continue firing on the outpost's position. However, if the RCC will know when it has eliminated a BC outpost, then it will not need to target that BC outpost any further.

4. Red firing doctrine: whether the RM force will fire only upon the closest BM force, or will divide its firepower equally among all BM forces within firing range.

5. Red also has the opportunity to select its route of attack and speed of advance. This could enable the Red force to stay out of BC sensor range, or only be within sensor range for a short time period and attack the BM force when it only has unaimed units.

The other variables which the user must decide on are

1. Weapon characteristics. The user must decide on maximum firing range for both Blue and Red weapons and the firing rate of those weapons.

2. Probability of Kill. The user must assign probability of kill's for the Blue unaimed units, Blue sensor-aimed units, and the Red units. (RM and RCC units have the same probability of kill.)

3. Hand-to-hand combat range. The user should determine a distance at which the opposing forces would no longer be able to fire their weapons at each other and would change over to hand-to-hand combat.

All of the above decisions can have a direct bearing on the outcome of the battle. By varying the above parameters and analyzing the results of battles, the user should be able to gain valuable insights into the effects of C2 and counter C2.

V. THE COMBAT MODEL

The previous section dealt with the C2 process and how it affects the forces involved. This section explains the combat portion of the computer simulation. It describes the attrition calculations for the forces, the differences between the deterministic and stochastic versions of the simulation, the battle termination and the Red force movement.

A. THE ATTRITION CALCULATIONS

There are two types of units represented in this simulation, unaimed and sensor-aimed units, and the attrition calculations are performed differently for each type of unit. However, when two forces are firing at each other, they are assumed to fire on each other simultaneously.

1. Unaimed Attrition Calculation

An unaimed force means there is no coordination among the firing units. All of the Red force (both RM and RCC units) and the Blue unaimed units (BU) have no coordination. The expected Red force attrition by Blue unaimed units during a game step is found using the following equation:

$$E(K)=RM\left[1-\left[1-(1/RM)\,(1-(1-BUPK*\exp(-RNG)))\right]^{BFR*STEP}\right]^{BU} \qquad (5.1)$$

where

E(K) = expected number of Red main units killed
by Blue unaimed (uncoordinated) fire

RM = the number of Red main units being fired upon

RNG = distance from the firing BU units to the
receiving RM units

BUPK = Blue unaimed probability of Kill

BFR = Blue firing rate in rounds per minute

STEP = number of minutes in a game step

BU = the number of firing Blue unaimed units.

The value of the variables RM, BUPK, BFR, STEP, and BU are user supplied during the input phase of the simulation. The value of RM and BU change during the course of the battle due to attrition and, in the case of BU, can also change due to the transformation from BU to BA.

Likewise, the expected Blue unaimed attrition by the Red Main units during a game step is found using the following equation:

$$E(K)=BU\left[1-\left[1-(1/BU)\,(1-(1-RPK*\exp(-RNG)))\right]^{BFR*STEP}\right]^{RM} \qquad (5.2)$$

where

E(K) = expected number of Blue unaimed units killed

by Red Main fire

BU = the number of Blue unaimed units being fired upon

RNG = distance from the firing RM units to the receiving BU units

RPK = Red probability of kill

RFR = Red firing rate in rounds per minute

STEP = number of minutes in a game step

RM = the number of firing Red Main units

The value of the variables RM, RPK, RFR, STEP, and BU are user supplied during the input phase of the simulation. The Blue sensor-aimed units (BA) attrition is found by subsituting the value BA for BU in equation (5.2). The BC outpost attrition is also found by using equation (5.2). The value of BC is subsituted for BU, RCC is subsituted for RM and the value of RNG is the distance from the firing RCC unit to the receiving BC outpost. The rationale for these expressions is given in Appendix B.

2. Aimed Attrition Calculation

The only aimed force in this simulation are the Blue sensor-aimed units (BA). The BA units are receiving targetting information from the BC outposts which increase their probability of kill.

The expected number of Red units killed by BA during a game step is represented as follows:

if BA/RM≤1, then

E(K) = BA*(1-(1-BAPK*exp(-RNG))**BFR*STEP) (5.3)

or if BA/RM>1, then

E(K) = RM*(1-(1-BAPK*exp(-RNG))**(BFR*(BA/RM)*STEP) (5.4)

where

E(K) = expected number of Red units killed by BA
BA = number of firing Blue sensor-aimed units
RM = number of RM units being fired upon by BA
RNG = distance from the firing BA units to the receiving RM units
BAKP = Blue sensor-aimed probability of kill
BFR = Blue firing rate in rounds per minute
STEP = number of minutes in a game step

Both the equations (5.3) and (5.4) assume that at the beginning of the game step each available Red target is acquired and there is no opportunity to check for the effect of a shot during the game step and change the aim point if there was a successful kill. However, checking for kill, and change of target, is possible at time step termination. (See Appendix B for the derivation of equations (5.1), (5.2), (5.3) and (5.4)).

B. DETERMINISTIC VS STOCHASTIC VERSIONS

The user has the option of selecting either a deterministic or stochastic version of the combat model. The user selects the version of the model by the variance variable (line number 4) in the Table of Variables and Current Values (see figure 2.2). If the user sets the variance equal to zero, the deterministic version of the model is used. If the variance is greater than zero, a stochastic combat model is in effect.

In either version of the simulation, an expected number of kills (E(K)) for each type of unit (RM, RCC, BU, BA, and BC) for every game step is calculated from equations (5.1), (5.2), (5.3), or (5.4). In the deterministic model, the expected number of kills (E(k)) for each game step is the attrition rate for the force. In the stochastic version, the expected number of kills (E(K)) is used as a basis for calculating the random attrition rate A^+ for the unit. The attrition rate A^+ is found from the following equations:

$$A^+ = \exp(U + S*Z) \qquad (5.5)$$

where

$$U = \ln(E(K)) - 0.5(S)^2 \qquad (5.6)$$

$$S^2 = VAR*\ln(1+1/E(K)) \qquad (5.7)$$

and

VAR = the variance as entered by the user

Z = is a random number from a normal (0,1) distribution

In fact, the stochastic attrition is governed by a log-normal model. (See Appendix C for a derivation of equations (5.5), (5.6), and (5.7)).

C. HAND-TO-HAND COMBAT

Hand-to-hand or "close-in" combat takes place when a Red force moves so close to a Blue force that neither of the forces can fire their weapons at one another. The outcome of hand-to-hand combat is determined by the number of forces each side has at the beginning of the "close-in" combat. The force with the largest number of units will win the battle. The winning force is reduced in strength by the number of units it fought against. For example, if a Red force had a total of 90 units (60 RM units and 30 RCC units) and a Blue force had a total of 60 units, then after the hand-to-hand combat, 30 Red units would have survived. (There would be 20 RM units and 10 RCC units.)

The hand-to-hand combat calculations are as follows: if the Red (RT) force outnumbers the Blue (BM) force (RT>BM), then

$$RM = (RT - BM)*RM/RT \tag{5.8a}$$

$$RCC = (RT - BM)*RCC/RT \tag{5.8b}$$

$$BM=BA=BU=0.0 \tag{5.8c}$$

and if the Red (RT) force equals or is outnumbered by the Blue (BM) force (RT≤BM), then

$$BU = (BM - RT)*BU/BM \tag{5.9a}$$

$$BA = (BM - RT)*BA/BM \tag{5.9b}$$

$$RT=RM=RCC=0.0 \tag{5.9c}$$

where

RT = total number of Red units in hand-to-hand combat

RM = the number of RM units in hand-to-hand combat

RCC = number of RCC units in hand-to-hand combat

BM = total number of Blue units in hand-to-hand combat

BU = the number of BU units in hand-to-hand combat

BA = the number of BA units in hand-to-hand combat

The simulation assumes that if a unit is engaged in hand-to-hand combat during a game step then it

1) Can not be fired upon by an opposing force during the game step;

2) The unit can not fire on any other opposing force during the game step.

The hand-to-hand combat range is a user supplied variable during the input phase of the simulation.

D. BATTLE TERMINATION

The battle is considered over when either all of the Red force or all of the Blue fighting force is annihilated. The forces can be eliminated either through attrition or hand-to-hand combat.

E. BREAK POINTS

The user can input a Red break point and a Blue break point. The break points are not utilized during the battle itself, but only after it is over, during the interactive graphics portion of the program. Break points indicate to the user at what point in time the forces would have broken off the engagement. The break points allow the user to examine what effect breaking off the engagement at that point in time would have had on the outcome of the battle.

F. FORCE MOVEMENT

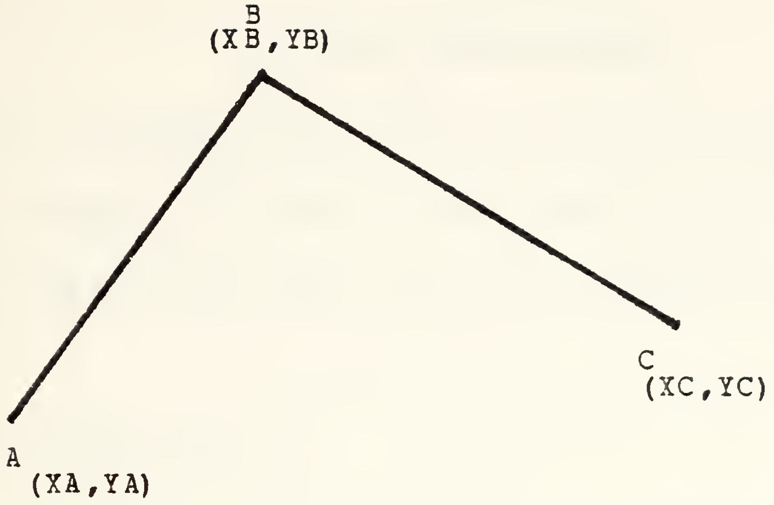
The present version of the simulation allows the Red units the option of movement. The Blue units (both BM and BC outposts) are stationary. The user can choose the position of the BM and BC outposts, but once these positions are selected, they remain fixed during the battle.

The user also selects the initial position of the Red units. However, unlike the Blue units, the Red units can

move across the battlefield in any direction and at a user supplied speed. The user can specify up to 13 turning points at which the Red units change direction. After the user has entered Red's final turning point, the program automatically finds the location of the closest Blue Main force and ensures that the Red unit will move to within hand-to-hand combat range of the Blue unit. This feature ensures the battle will terminate.

The method used to calculate the distance a Red unit travels and its location is as follows. First, the distance the Red unit would move during a game step is found using equation (5.10). Then the distance between the Red unit (point A in Figure 5.1) and the next Red turning point (point B in Figure 5.1) is calculated using equation (5.11). If the distance is less than the distance calculated by equation (5.10), the new coordinates of Red's location are found using equations (5.12), (5.13), and (5.14). However, if Red would have travelled beyond point B, the distance from point A to point B is subtracted from the distance found in equation (5.10) and the Red unit is moved to point B. The distance from point B to point C is then calculated (equation (5.11)) and compared to the distance the Red unit

has left to move. Again, if the distance Red has remaining to move is less than the distance between point B and point C, the new coordinates of Red's location are found using equations (5.12), (5.13), and (5.14). This process continues until the Red unit has reached its final destination.



$$DST = SPEED * TIME \tag{5.10}$$

$$LEN = \sqrt{(XA - XB)^2 + (YA - YB)^2} \tag{5.11}$$

$$ANG = TAN^{-1} (|YB - YA| / (XB - XA)) \tag{5.12}$$

$$Xf = XA + LEN * COS(ANG) \tag{5.13}$$

$$Yf = YA + LEN * SIN(ANG) \tag{5.14}$$

where

Xf and Yf are the final coordinates of the Red unit

DST = distance a Red unit travels during a game step

SPEED = Red speed of advance

TIME = number of minutes in a game step

Points A, B, C are user specified turning points

Figure 5.1 Red Movement Equations.

VI. FUTURE ENHANCEMENTS

After completing the present version of the program, it became obvious that several enhancements could be made. Some of these improvements will be discussed in this section.

A. TWO-PLAYER GAME

The simulation could be made into a true two-player game. In a limited sense, the present version can be considered a two player game since their are two opposing forces and one player could make the Blue decisions and the other player could make the Red decisions. However, it would be a much better game and learning experience if both forces had the opportunity to have the same type of units, ie, Red with command and control outposts and Blue with counter-command and control forces. By allowing both the Red and Blue forces to have equal capabilities, the combat outcome would depend on how effectively a player allocated his resources, where he initially positioned his forces, and the strategy the player followed during the course of the battle.

In a true two-player game, the concepts of secrecy and information hiding could be incorporated. The information displayed to each player could be made dependent on his sensor inputs. In this way, each player would be making decisions based on information obtained from his sensors and the accuracy of that information could be a function of the amount of resources devoted to his sensor system and the amount of resourses his opponent allocated to counter-command and control.

B. RE-SUPPLY OF FIGHTING FORCES

A further improvement would be to give both the Red and Blue fighting forces an opportunity to re-supply. This capability would enable the user to gain insight into how a battle could be won or lost through a timely decision on when and where to re-enforce the fighting units.

This improvement could be made to the present version of the simulation and it could be incorporated into the two-player game.

C. NEW TRANSFORMATION EQUATIONS

A user might also want to develop new, or refine the present equations for transforming BU to BA units. It would be interesting to explore the sensitivity of battle results to changes in transformation equations.

In the present simulation, the BA units have a higher probability of kill than the BU units. A new transformation might possibly increase the firing rate, or decrease the attrition rate of the BA units by RM units, or be a combination of the three variables--probability of kill, firing rate, and attrition rate of BA.

In this computer program almost any combination is possible since all the variables are user defined and the program's structure is modular. The user simply writes one routine representing his new transformation equation and inserts it in place of the old transformation routine.

D. DEVELOPMENT OF COMBAT SCENARIOS

A fourth enhancement would be the development of different combat scenarios to be played out in the simulation. The new scenarios might incorporate the forest and cities hexes which are on the playing board but not presently in use, or they might place restrictions on the two forces.

Having a variety of scenarios would allow the user to investigate the effect of command and control in different situations. It would give the user the opportunity to gain insights into which parameters are more important (have a greater effect) in the different situations. For example,

in what kind of combat situation is the decision time of a C2 system most important. One combat situation might be a surprise attack, while another would be one in which the sensors detect the approaching opposing forces before they opened fire. Or, if the restriction on the Blue force was that for every minute under a seven minute command and control decision time the fighting force was reduced 50 units, then how many fighting units would the Blue commander be willing to give up in order to gain the extra time and supposedly higher transformation rate for his unaimed units. If the Blue commander knew that the opposing force could never mount a surprise attack, then the extra time might not be needed. In fact, it might be decided to increase the C2 decision time to 8 or 9 minutes and thereby gain extra fighting forces.

Many different scenarios could be investigated to determine the sensitivity of battle results to C2 decisions.

APPENDIX A

USER'S MANUAL

A. INTRODUCTION

This computer simulation demonstrates the effect of a Command and Control (C2) system on a stylized simple combat situation using a simple C2 model. The program is interactive and user-friendly utilizing high-resolution color graphics to display intermediate and end-of-battle results.

It is hoped that use of this simulation will provide a user with insights into the effect of C2 upon the outcome of conflicts. The user, by varying the amount of C2 and counter C2 used during a particular battle, should come to understand and appreciate the importance of C2, and how C2 decisions can easily change the outcome of a battle.

The purpose of this manual is to familiarize the user with the simulation and user inputs necessary to run the simulation.

B. PROGRAM STRUCTURE

The simulation is a modular, menu driven program coded in FORTRAN. The software can be divided into five distinct sections:

1). the main program

2). the input section

3). the combat section

4). the command and control section

5). the graphical output section.

A brief description of all five sections and their major subroutines is included in this manual to aid a user in understanding the simulation and to allow a user to add new, or to improve on the old subroutines.

1. The Main Program

The main program, along with the BLOCK DATA module, has several important functions. It initializes the RAMTEK 9400 to accept the graphcial output, calls the major input subroutine, the major combat subroutines, the command and control subroutines, and the output subroutines for both the intermediate and end-of-battle displays.

The simulation uses common statements to pass parameters from subroutine to subroutine. The common statements are broken down so that only the variables used by a subroutine are included in that subroutine.

The general flow of the program is shown in the flowchart in Figure A.1.

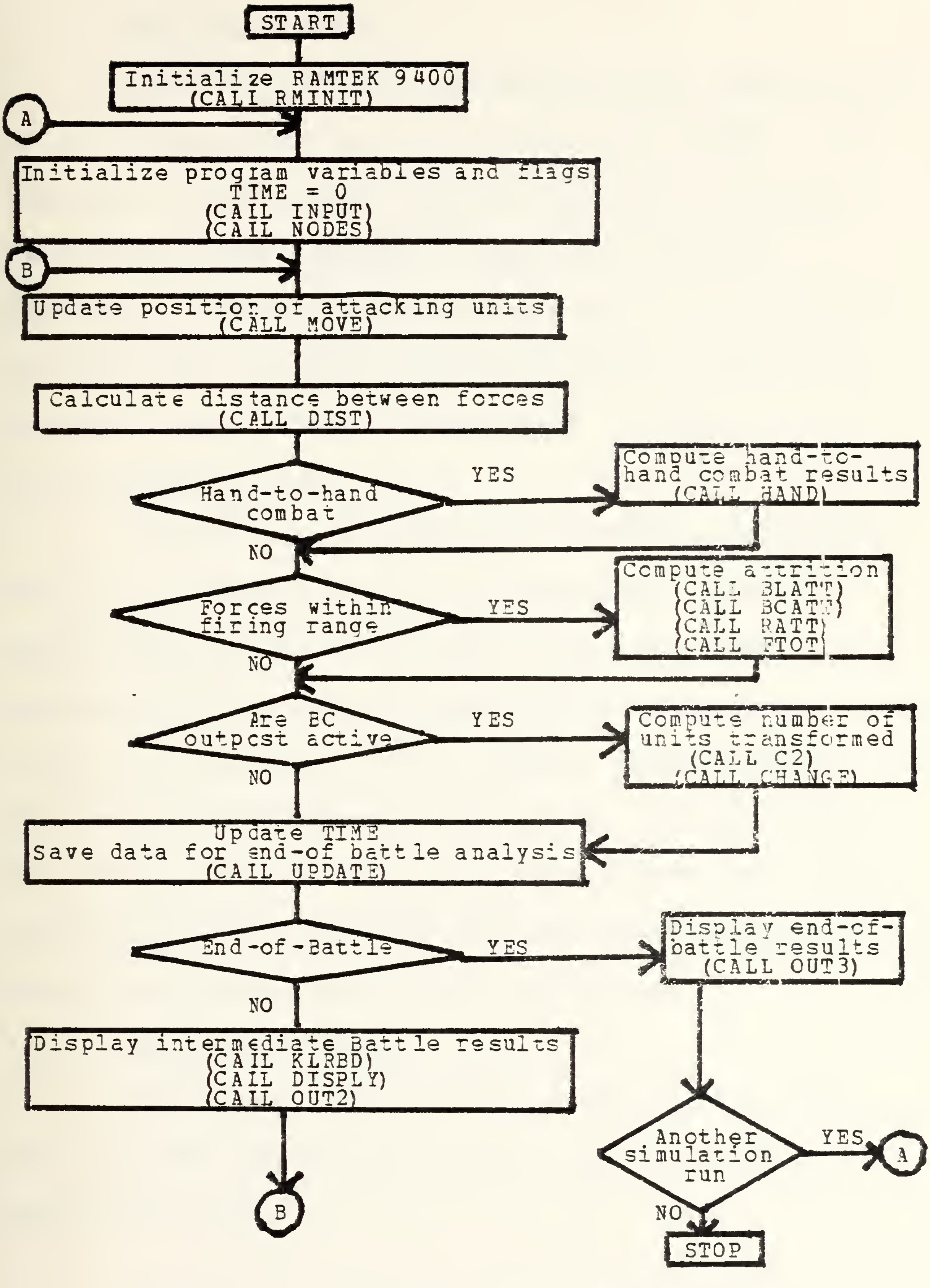

START
Initialize RAMTEK 9400 (CALL RMINIT)
A
Initialize program variables and flags
TIME = 0
(CALL INPUT)
(CALL NODES)
B
Update position of attacking units (CALL MOVE)
Calculate distance between forces (CALL DIST)
Hand-to-hand combat
YES
Compute hand-to-hand combat results (CALL HAND)
NO
Forces within firing range
YES
Compute attrition
(CALL BLATT)
(CALL BCATT)
(CALL RATT)
(CALL FTOT)
NO
Are BC outpost active
YES
Compute number of units transformed
(CALL C2)
(CALL CHANGE)
NO
Update TIME
Save data for end-of battle analysis
(CALL UPDATE)
End-of-Battle
YES
Display end-of-battle results (CALL OUT3)
NO
Display intermediate Battle results
(CALL KLRBD)
(CALL DISPLY)
(CALL OUT2)
B
Another simulation run
YES
A
NO
STOP

Figure A.1   Flow Chart of the Main Program.

2. The Input Section

There are two major subroutines in the input section of the program. The first is the subroutine INPUT. INPUT displays the menu titled 'Table of Variables and Current Values' and allows the user to change any variable displayed. INPUT provides explanations of all the program's variables and, if a user decides to change the value of a variable, it will insure the new value is within the variable's specified range.

The second major subroutine is NODES. NODES first displays the playing board, then draws the approximate sensor range for each BC outpost on the playing board, and finally allows the user to enter the Red forces' route of attack. The user must use the RAMTEK's trackball when designating the Red's units attack route. After the user has entered the route of attack, NODES ensures that the Red force will come to within hand-to-hand combat range of a Blue force, thus assuring battle termination.

3. The Combat Section

The combat section includes force attrition, Red troop movement, hand-to-hand combat, and force strength update routines.

Force attrition calculations are done in three subroutines. RATT (Red attrition) calculates the expected number of Red units killed by the Blue units. BLATT (Blue attrition) calculates the expected number of Blue Main units killed by the Red force. The third subroutine BCATT (BC outpost attrition) calculates the expected loss of BC resources to RCC fire. In the routine FTOT, the true attrition rate for the forces is calculated. If the deterministic model is in effect, then the true attrition rate is set equal to the expected number of kills calculated in RATT, BLATT, and BCATT. However, if the stochastic version of the model is in effect, then the true attrition rate is calculated using a log-normal distribution.

The Red troop movement is performed by the routine MOVE. The routine MOVE moves the Red force along the route selected by the user. The routine HAND calculates the results of all hand-to-hand combat. And finally, the routine UPDATE saves the data needed by the routines which display the end-of-battle results.

4. The Command and Control Section

There are three main routines used by the command and control section. The three routines are DIST, C2, and

CHANGE. DIST calculates the distance between all the units on the playing board. The range between units is used in all attrition calculations, and also in the transformation calculations. For each BC outpost, the routine C2 calculates the maximum number of BU units to be transformed to BA units during a game step. The routine CHANGE takes the output of the C2 routine and for each Blue Main force calculates the actual number of BU units transformed to BA units.

5. The Graphical Output Section

There are two distinct graphical output displays. The first is the intermediate battle results which are displayed at end of every game step. This display is built be several subroutines. The subroutines include BOARD, PUTMEN, DELETE, DISPLY, LTRSZE, CLRBD, and several minor routines. The force status reports shown at the bottom of the display are generated by the routines OUT1 and OUT2.

The second graphical output is the Force Strength vs Time plots. These graphs are generated by the routine OUT3 and displayed after combat terminates. OUT3 calls several minor routines which draw the axis (BAXIS), scales the data (BSCALE), draws the break points for the forces (BRKPT), and connects the data points with a continuous colored line (BLINE).

C. THE INPUT DATA

Prior to running the simulation, the user must initialize the variables listed in the menu titled 'Table of Variables and Current Values'. To aid the user all the variables have been given default values. To change a variable's initial value, or to obtain an explanation of the meaning of the variable, the user simply enters the line number associated with that variable. For example, the line number associated with hand-to-hand combat range is 6. After entering a 6, the user would receive an explanation of hand-to-hand combat range and then have the option of either changing its present value, or leaving it unchanged.

The following is a list of all the variables, their associated line numbers, default values, and an explanation of the meaning of the variable.

Line Number	Variable Default Value	Explanation of Variable
1	Number of Red Forces Default: 2	The number of active Red forces. Each Red force consists of RM and RCC forces. There can be either 1 or 2 Red forces.
2	Number of Blue Forces Default: 2	The number of active BM forces. Can have 1, 2,

		or 3 forces.
3	Number of Blue Sensor Outposts (BC) Default: 2	The number of active BC outposts. Can have 0 to 6 BC active during a battle
4	Seed Default: 12345	Used to generate a random number. Should be an integer from 0 to 99999.
	Variance Default: 1.0	If variance=0.0, then a deterministic model is in effect. If variance>0.0, then stochastic model is in effect. The variance indicates the magnitude of variation in the attrition rate. The range is between 0.0 and 10.0.
5	Time Step (Min) Default: 10.0	The duration in minutes of each engagement. Range should be from 1.0 to 99.0 minutes.
6	Hand-to-Hand Combat Range (KM)	Distance at which forces change to hand-to-hand

	Default: 0.25	combat. The range should less than 1.0 KM.
7	Red Firing Rate (rounds/min) Default: 5.0	Number of shots fired per shooter per minute by the. RED forces. Range should be from 1.0 to 10.0 shots.
8	Red Maximum Firing Range (KM) Default: 3.0	Distance at which Red forces open fire. Range should be from 0.0 to 10.0
9	Red Probability of Kill Default: 0.40	Probability of kill for each shot fired. Range should be from 0.0 to 1.0.
10	Red Speed of Advance (KPH) Default: 3.0	Speed at which Red forces move across the battlefield. Range should be from 1.0 to 10.0 KPH.
11	Red Main Split Firing Allowed Default: YES	If YES, then RM units will fire on all targets within range. If NO, then RM will only fired on closest BM force.
12	RCC Split Firing Allowed	If YES, then RCC units will fire on all BC units

	Default: YES	within range. If NO, then RCC will fire on only closest BC outpost.
13	RCC will fire at only non-zero BC outpost Default: NO	If YES, then RCC will only fire at BC outpost which have not been eliminated, BC which have resources greater than zero. If NO, then RCC will fire at all identified BC outpost. Even those which have had their resources reduced to zero.
14	Red Men and Location Default: RM1 = 900, RCC1 = 900, X = 8.0, Y = 3.0	The number of RM and RCC units initially active in the first Red force and their initial location. RM and RCC should from 0 to 10000. The X and Y coordinates should be between 0.0 and 10.0.
15	Red Men and Location Default: RM2 = 900,	The number of RM and RCC units initially active in

	RCC2 = 100, X = 7.5, Y = 7.5	the second Red force and their initial location. RM and RCC should from 0 to 10000. The X and Y coordinates should be between 0.0 and 10.0.
16	Red Break Point Default: 0.70	The percent of losses of Red's initial strength at which Red would retreat Should be less than 1.0.
20	Blue Firing Rate (rounds/min) Default: 5.0	Number of shots fired per shooter per minute by the Blue forces. Range should be from 1.0 to 10.0.
21	Blue Maximum Firing Range (KM) Default: 3.0	Distance at which Blue force open fire. Range should be from 1.0 to 10.0.
22	Blue Speed of Advance Default: 0.0	Speed at which the Blue forces move. Not used in version of the simulation.
23	Blue Unaimed Probability of Kill Default: 0.25	The probability of kill per shot of the BU units. Must be from 0.0 to 1.0.

24	Blue Sensor-Aimed Probability of Kill Default: 0.70	The probability of kill per shot of the BA units. Must be from 0.0 to 1.0.
25	Blue Split Firing Allowed Default: YES	If YES, then BM units will spread its fire equally among all Red within range. If NO, then BM units will fire on only the closest Red force.
26	Blue C2 Decision Time (MIN) Default: 5.0	The time required by the Blue forces to respond to the Red hostile action. The range should be from 0.0 to 99.9 minutes.
27	Blue Men and Location Default: BU1 = 800, X = 4.0, Y = 4.0	The number of Blue unaimed units initially active. BU should be less than 10000. The X and Y coordinates should be between 0.0 and 10.0.
28	Blue Men and Location Default: BU2 = 800,	Same as line 27.

	X = 3.5, Y = 6.5	
30	Blue Break Point Default: 0.50	The percent of losses of Blue's initial strength at which Blue would retreat. Should be less than 1.0.
40	Red Range Weight Default: 2.0	The importance of the distance from the Red force to the reporting BC outpost. Should range between 1.0 and 10.0 where 1.0 means the most important factor and 10.0 means the least important factor
41	Blue Range Weight Default: 10.0	The importance of the distance from the Blue force to the reporting BC outpost. Should range between 1.0 and 10.0 where 1.0 means the most important factor and 10.0 means the least important factor
42	BC outpost Force,	For each BC outpost the

to 47	Sensor Range, Time on per Game Step, and Position Default: BC1 = 300, Sensor Range = 3.0, Time on = 1.0, X = 5.5, Y = 4.5 BC2 = 300, Sensor Range = 3.0, Time on = 1.0, X = 6.0, Y = 6.0	initial amount of resources (force) allocated to the outpost, its maximum sensor range, the percentage of a game step the outpost is active and its position. Initial unit strength should be less than 9999. Maximum sensor range should be between 1.0 and 10.0. Time on should be between 0.0 and 1.0. The X,Y coordinates should be between 0.0 and 10.0.

The user indicates to the program that he has finished initializing the variables by entering a 99 at the keyboard. After entering 99, the user can select the Red route of attack. Entering the attack route has been simplified for the user through the use of the RAMTEK's trackball. The user designates all turning points by moving the trackball cursor to the hex where the turning point is to occur and

depressing the enter key on the trackball. The user can designate a maximum of 13 turning points for each Red unit. The Red force will travel in a straight line from turning point to turning point. When the desired path of the Red force has been entered then move the cursor into the red area at the bottom of the screen and depress the enter key on the trackball.

On the second and all subsequent runs, the user is given the option of entering a new route of attack or using the previous route of attack. If the user selects to use the previous route of attack, the Red force will start and finish at the same location as in the last run. Therefore, the user should not change the X or Y coordinates of either the Red force or the Blue force when electing to use the attack route of the previous run.

D. OUTPUT

There are two distinct high-resolution graphical displays used by the simulation. One is the playing board displays which indicate unit position and strength at the end of each game step and provide one of four different status reports at the bottom of the screen. The other is the Force Strength vs Time plot which is available to the user at the end of a conflict.

1. Playing Board Displays

Figures 2.5, 2.6, 2.7, 2.8 are examples of playing board displays. Each picture shows the same point in time of a battle but a different status report is displayed at the bottom of the screen. The user selects the status report desired by placing the trackball cursor over the letters representing the status report and depressing the enter key. For example, to view the Blue Force Status report the cursor is placed over the word 'BLUE' and the enter key is depressed. To examine the BC Outpost Status report the cursor is placed over the letters 'BC' and the enter key is depressed. When the user places the cursor over the word 'MODEL' and depresses the enter key, the position and strength of all the forces for the next game step is displayed on the playing board.

2. Force Strength vs Time Plots

At the end of a battle, the user can build graphs such as the one shown is Figure 2.9. To build a graph, the use selects a force to be plotted. For example, to plot the Blue total fighting force strength (BT1) vs time, the user places the cursor over the letters BT1 and depresses the enter key. A plot of force strength vs time will be made

for the first Blue Main force. Spaced along the line will be the letter in parenthesis next to the force selected, in this case the letter G. This feature allows the user to distinguish between lines of the same color. (The Red forces have red lines and letters; the Blue fighting forces have blue lines and letters; the BC outposts have black lines and letters.) Break points (a user input) are shown on the total force strength lines only (RT1, RT2, BT12, BT2, BT3). Break points enable the user to hypothesize what would have happened had the units retreated upon reaching their break point.

To clear the screen and replot a force, the user places the cursor over the words 'NEW PLOT' and depresses the enter key. To terminate the plotting routine and return to the beginning of the next simulation run, the user places the cursor over the word 'RETURN' and depresses the enter key.

There is a maximum of 50 data points per line which can be plotted in this routine. Therefore, if a battle simulation has more than 50 game steps, only the first 50 data points will be plotted.

E. TO RUN THE SIMULATION

The simulation program is located in Spanegle Hall room 500 on the PDP 11/50 computer system. To run the simulation, the user must do the following:

1). Turn on the RAMTEK 9400. The ON/OFF switch is located on the front panel of the RAMTEK 9400 cabinet.

2). Turn on the RAMTEK 9400 keyboard, trackball, and color monitor. All of the ON/OFF switches for these devices are clearly marked on the front of each device.

3). Log on to the PDP 11/50 system at any terminal located in the room. The commands are

```
>HEL BENT
>password:  BENT
        (a system message is displayed)
>RUN BENT
```

(Note that the > sign is a system prompt and that the user only types the letters that are capitalized. The lower case letters are displayed to the srceen by the PDP 11/50 operating system.)

4). After entering RUN BENT, the user must use the RAMTEK keyboard and trackball to communicate with the simulation. The program is interactive with step-by-step directions provided for the user.

5). After the last simulation run, the user should log off the system by entering the following command at the terminal at which he logged on

```
>BYE
```

To obtain a copy of the source code, the user should enter the following command prior to logging off the system

```
>@PRNT
```

APPENDIX B

'DERIVATION OF ATTRITION EQUATIONS'

Suppose a Blue force confronts a Red Force. Further, suppose that Blue attacks Red and does so without coordination (command and control), ie, each B in Blue picks a member R in Red at random and fires at it independently of the actions of the other B's in Blue. Assume that all Rs in Red are equally likely to receive fire from a single B. Also, assume that the kill probability of B against R is given by the expression

$$Pb = Pw * \exp(-RANGE) \qquad (B.1)$$

where RANGE is the distance between the firing B and the receiving R and Pw is the probability of kill for the weapon fired. Further, let γ denote the firing rate of B.

Let K denote the random number of Rs killed by the Bs per engagement (of duration λ). Then during the engagement of time λ, the B fires $\gamma * B$ shots. The number of Rs killed, K, may be expressed as

$$K = I(1) + I(2) + \ldots + I(j) + \ldots + I(r) \qquad (B.2)$$

where R is the number of Reds receiving fire during the period, and the indicator random variable

$$I(j) = \begin{cases} 1 & \text{if the jth R is killed by B fire} \\ 0 & \text{otherwise} \end{cases}$$

then the expected number of kills, E(k), is

$$E(K) = \sum_{j=1}^{R} E(I(j)) = R*E(I(1)) = R*P(I(j)=1) \qquad (B.3)$$

where P(I(j))=1) is the probability of at least one kill during the engagment duration λ, assumed to be the same for each element of the Red force. If E(I(i)) can be computed, then E(K) can be found.

First of all, it should be noted that the probability that i of the B's target the jth R is given by the binomial

$$\binom{B}{i} (1/R)^{i} (1-(1/R))^{B-i} \quad , \quad i=0,1,2,\ldots,B \qquad (B.4)$$

Then, given that a B is targeted on the jth R, the probability of at least one kill in an engagement of duration is

$$1 - (1 - Pb)^{\gamma\lambda}$$

Therefore, the probability of at least one kill by i B's firing independently is

$$1 - ((1-Pb)^{\gamma\lambda})^{i} = 1 - Z^{i} \qquad (B.5)$$

where Z is the probability of no kills per engagement with one B. Now,

$$P(I(j)=1) = \sum_{i=0}^{B} (1- Z^{i}) \binom{B}{i} (1/R)^{i} (1 - (1/R))^{B-i}$$

$$= 1 - (Z * (1/R) + 1 - (1/R))^{B} \qquad (B.6)$$

Since E(I(i)) = P(I(i)=1), then

$$E(I(i)) = 1 - (1-(1/R)*(1-(1-Pb)^{\gamma\lambda})^{B} \qquad (B.7)$$

Then from equation (B.3),

$$E(K) = R*(1 - (1 - (1/R)*(1- (1-Pb)^{\gamma\lambda}))^{B} \qquad (B.8)$$

Equation (B.8) is Equation (4.2) which is used in the computer simulation to find the expected number of kills from unaimed fired.

When B is firing on R with coordination, the experssion for E(K) becomes

$$E(K) = B*(1 - (1-Pb)^{\gamma\lambda}) \qquad \text{for } B/R \leq 1 \qquad (B.9)$$

and

$$E(K) = R* \left(1 - (1-Pb)^{\gamma\lambda(B/R)}\right) \qquad \text{for } (B/R) > 1 \qquad (B.10)$$

Equation (B.9) assumes that one B per R is allocated for as long as the B's last. Equation (B.10) assumes that (B/R) B'S are allocated to each R during the engagement. No account is taken of the fact that B/R is often not an integer.

Equation (B.9) and (B.10) are the aimed attrition equations of Chapter IV.

APPENDIX C

DERIVATION OF THE STOCHASTIC ATTRITION RATE

Assume that F(t-1) represents the size of the Red force at the (t-1) step in time. Let A^{+} be the stochastic attrition rate of the Red force by the Blue force at the t step in time. Let Or(t) be equal to the expected number of kills (E(k)) of the Red force by the Blue force found by using equations (5.1), (5.3), and (5.4). An equation for A^{+} will now be developed based on a log-normal distribution.

For a log-normal distribution

$$X = \exp\ (U + S*Z) \tag{C-1}$$

where

Z is a random variant from a normal (0,1) distribution

U is the mean

S is the standard deviation

The expected value of X is

$$E(X) = E(\exp(U + S*Z))$$

$$= \exp(U) * E(\exp(S*Z))$$

but $E(\exp(\phi*Z)) = \exp(0.5*\phi^2)$

Therefore,

$$E(X) = \exp(U + 0.5*S^2) \tag{C-2}$$

If the expected value of X, E(X), is equal to the expected number of kills $\theta r(t)$ of the Red force, then

$$E(X) = \theta r(t)$$

and

$$\theta r(t) = \exp(U + 0.5*S^2)$$

$$\ln(\theta r(t)) = U + 0.5*S^2$$

and therefore,

$$U = \ln(\theta r(t)) - 0.5*S^2 \qquad \text{(C-3)}$$

The variance of X is

$$VAR(X) = E(X^2) - (E(X))^2$$

Now $$E(X^2) = \exp(2U + 2S^2)$$

and $$(E(X))^2 = \exp(2U + S^2)$$

Therefore,

$$VAR(X) = \exp(2U + 2S^2) - \exp(2U + S^2)$$

$$= \exp(2U + S^2)*(\exp(S^2) - 1)$$

$$= (E(X))*(\exp(S^2) - 1)$$

For a Poisson Distribution,

$$VAR(X)/E(X) = 1$$

Therefore, assuming Poisson variability in kills,

$$1 = E(X)*(\exp(S^2) - 1)$$

and if $\theta r(t) = E(X)$, then

$$1 = \theta r(t)*(\exp(S^2) - 1)$$

$$(\exp(S^2) - 1) = 1/\theta r(t)$$

$$S^2 = \ln(1 + 1/\theta r(t)) \qquad \text{(C-4)}$$

and when $\quad VAR(X)/E(X) \neq 1$

then

$$S^2 = VAR(X)*(\ln 1 + 1/\theta r(t)) \qquad \text{(C-5)}$$

and

$$S = SQRT(S^2) \qquad \text{(C-6)}$$

Finally, let A^+ be equal to X in equation (C-1) and then

$$A^+ = \exp(U + S*Z) \qquad \text{(C-7)}$$

where

U is the mean found in equation (C-3)

S is the standard deviation found in equations (C-5) and (C-6) and

Z is a random variate from a normal (0,1) distribution

BIBLIOGRAPHY

Gaver, Donald, P. "Models that Reflect the Value of Information in a Command and Control Context." Naval Postgraduate School Technical Report NPS-55-80-027.

Lawson, J, S. "The Role of Time in a Command Control System". Naval Electronics System Command. Washington, D.C.

Lawson, J, S. "Command Control as a Process". Naval Electronics System Command. Washington, D.C.

RM-9400 FORTRAN Interface Package (FIP). Santa Clara, 1980.

Tonguc, K. *Modeling the Effect of Information on Conflict Outcome.* M.S. Thesis, Naval Postgraduate School, Monterey, California, 1979.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor D. P. Gaver, Code 55Gv Department of Operations Research Naval Postgraduate School Monterey, California 93940	2
4.	Capt J. E. Bent HQ ECD/DCXRC APO New York 09012	2
5.	Professor J. G. Taylor, Code 55Tw Department of Operations Research Naval Postgraduate School Monterey California 93940	1
6.	Professor J. M. Wozencraft, Code 62Wz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
7.	Professor P. H. Moose, Code 62Me Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
8.	CDR Gary Porter, Code 55Pt Director C2 Warlab Naval Postgraduate School Monterey, California 93940	1
9.	CDR Adams Center for War Gaming Naval War College Newport, Rhode Island 02840	1
10.	Naval Electronics System Command Dr. J. Lawson Washington, D.C. 20360	1
11.	Commander-in-Chief Pacific Command, Code 577 Camp Smith, Hawaii 96861	1
12.	U.S. Army War College Capt Howard Yellen, USA Carlisle Barracks, Penn 17013	1
13.	Professor M. Sovereign Chairman, Code 74 C3 Academic Group Naval Postgraduate School Monterey, California 93940	1

200616

16

Thesis
B3917 Bent
c.2 A computer simulation of a combat model which uses command and control.

on
ch

24 APR 87 32046

Thesis
B3917 Bent
c.2 A computer simulation of a combat model which uses command and control.

200616

thesB3917
A computer simulation of a combat model
3 2768 001 03734 4
DUDLEY KNOX LIBRARY